

Departamento de Ingeniería del Software e Inteligencia Artificial

Proyecto Fin de Máster en Ingeniería Informática para la Industria

Máster en Investigación en Informática

Facultad de Informática

Universidad Complutense de Madrid



**Sistema de reconocimiento y clasificación de agentes
patógenos de Nosemosis**

Claudia Marcela Alvarez Ramos

**Matilde Santos Peñas
Aránzazu Meana**

Madrid, Septiembre de 2010

AGRADECIMIENTOS

Agradezco a la Dra. Aránzazu Meana (Departamento de Sanidad Animal, Facultad de veterinaria de la universidad complutense de Madrid) por su asesoramiento científico y a su equipo de investigadores por la ayuda técnica en la manipulación y análisis de la información muestreada y así como al Dr. Mariano Higes del centro agrario de Marchamalo de la Junta de Comunidades de Castilla-la Mancha, por habernos proporcionado las muestras de N. Apis y N. Ceranae.

A mi querida tutora, la Dra. Matilde Santos Peñas por los valiosos consejos que me ha brindado a nivel profesional y personal, además de la gran dedicación que me ha brindado. Por su calidad humana y tantas cosas que no podría explicar, me siento honrada de haber trabajado a su lado.

Por último y no menos importante a mi familia funcional y disfuncional, su entrega, cariño y cooperación me inspiraron a seguir adelante para culminar mi meta. A mis amigas Silvia y María José, por haberme acogido como una hermana, ellas son fuente de inteligencia, fuerza, dedicación y armonía, a Andrés por ser mi amigo fiel.

A mis padres Yolanda y Joselyn, que cosieron mis alas y me ayudaron a volar de nuevo, por la inmensa fortaleza y amor que me transmitieron día tras día, no sería la persona que soy sin ellos. A mi hermana que en ella encuentre la paz que hace mucho tiempo me negué, mi ángel y a ti cariño, por acompañarme, apoyarme y creer en mí. Este proyecto es en vuestro honor.

RESUMEN

En los últimos años, el análisis de imágenes microscópicas es cada vez más importante tanto para la generación de diagnósticos como en la investigación en las Ciencias Naturales y de la Salud.

Actualmente existen herramientas robustas para el procesamiento de estas imágenes. Sin embargo, su gran capacidad es restringida por falta de adecuación a la problemática concreta. En esta investigación, se propone un método simple y directo de clasificación, medición y reconocimiento de esporas usando visión artificial, relacionada con los ámbitos de la Visión por Computador.

El método consiste en el desarrollo de un plugin para la plataforma *ImageJ* que automatice el proceso de clasificación de esporas desde el procesamiento de la imagen hasta la medición de la longitud y la anchura de las partículas, dando como resultado un reporte con la clasificación de las esporas analizadas. Así mismo, este resultado sirve para ayudar a diagnosticar las enfermedades parasitarias.

Palabras Clave: Diagnostico enfermedades parasitarias, Esporas, Plugin, Procesamiento de imágenes biomédicas, visión artificial, visión por computadora.

ABSTRACT

In the past few years, the microscopic image analysis is increasingly important for the generation of diagnostic and research in natural and Health sciences.

Currently there are robust tools for processing these images. However, its large capacity is constrained by lack of adaptation to the specific problem. In this research, we propose a simple and direct method of classification, measurement and recognition of spores using artificial vision, related to the fields of Computer Vision.

The method consists in the development of a plugin for the ImageJ platform that automates the process of classification of spores from the image processing to measure the length and width of the particles, resulting in a report with the classification of spores analyzed. Also, this result serves to help diagnose parasitic diseases.

Keywords: Vision, Parasitic diseases' diagnosis, Spores, ImageJ, Image processing, Plugin, Computer vision.

TABLA DE CONTENIDOS

AGRADECIMIENTOS	2
RESUMEN	3
ABSTRACT	4
TABLA DE CONTENIDOS	5
ÍNDICE DE FIGURAS	8
ÍNDICE DE TABLAS.....	10
1. Introducción	11
<i>1.1 Motivación.....</i>	<i>11</i>
Objetivos Generales	12
Objetivos Específicos.....	12
<i>1.2 Estructura del trabajo</i>	<i>13</i>
2. Estado del Arte	14
<i>2.1 El problema del despoblamiento de colmenas</i>	<i>14</i>
Síndrome de despoblamiento de las Colmenas (SDC).....	14

Detección de la enfermedad mortal “Nosemosis C”	15
Control de las enfermedades parasitarias	16
Métodos de Diagnóstico	16
2.2 <i>Algoritmos de procesamiento de imágenes</i>	18
2.2.1 <i>Eliminación de Ruido</i>	18
Filtro Gaussiano	19
2.2.2 <i>Conversión a escala de gris</i>	19
2.2.3 <i>Transformación Lineal</i>	20
2.2.4 <i>Ecualización del histograma</i>	20
2.2.5 <i>Algoritmos de Segmentación</i>	22
Algoritmos de similitud: Técnicas de umbralización	22
Algoritmos de discontinuidad: Sobel y Canny	23
2.2.6 <i>Detección de Elipse</i>	24
Algoritmo para calcular los parámetros de la elipse de una región	24
2.2.7 <i>Reconocimiento de patrones</i>	28
2.2.8 <i>Herramientas y librerías de procesamiento de imágenes</i>	28
ImageJ	28
IpLab (aForge.net)	30
3. Metodología de análisis, clasificación y diagnóstico	31
3.1 <i>Adquisición de la Imagen</i>	33
3.2 <i>Pre-procesamiento de la imagen</i>	34
3.3 <i>Segmentación</i>	36
3.4 <i>Reconocimiento de formas y dimensiones</i>	37
3.5 <i>Clasificación</i>	40
4. Discusión de Resultados	41
4.1 <i>Preprocesamiento de la imagen</i>	41
4.2 <i>Segmentación</i>	44
4.3 <i>Reconocimiento de Formas y dimensiones</i>	47

4.4 <i>Clasificación y Resultados</i>	48
5. Conclusiones y futuros trabajos	51
ANEXO I. TABLAS DE RESULTADOS	53
BIBLIOGRAFIA	55
RENUNCIA	58

ÍNDICE DE FIGURAS

Figura 2.1. Diagnostico en colmenas.....	14
Figura 2.2. N. Ceranae (a) y N. Apis (b).....	16
Figura 2.3. Imágenes microscópicas esporas de N. Apis (a) y N. Ceranae (b).....	17
Figura 2.4. Vistas detalladas de las células epiteliales del ventrículo parasitado a los 7 día, a una temperatura de 33 ° C.....	17
Figura 2.5. Histograma original - histograma ideal.....	21
Figura 2.6. Histograma Original- Histograma acumulativo.....	21
Figura 2.7. Clasificación de dos categorias del histograma.....	22
Figura 2.8. Anatomía de la elipse.....	25
Figura 2.9. Selección de tres puntos aleatorios en 3 de las 4 secciones de la elipse.....	25
Figura 2.10 Rectas tangentes puntos aleatorios seleccionados	26
Figura 2.11. Bisectriz de los puntos de intersección tangencial.....	27
Figura 2.12. Estructura simplificada arquitectura del software ImageJ.....	29
Figura 3.1. Diagrama global del tratamiento digital del conjunto total de imágenes.....	31

Figura 3.2. Imágenes digitales de las muestras (a) N. Ceranae con coloración (b) N. Ceranae sin coloración (c) N. Apis con coloración.....	34
Figura 3.3. Imagen en escala de grises NSEG8 y su histograma.....	35
Figura 3.4. Implementación en código java de la ecualización del histograma lineal.....	35
Figura 3.5. Operaciones Morfológicas.....	37
Figura 3.6. Eliminación de píxeles aislados.....	39
Figura 4.1. Aplicación filtro gaussiano.....	42
Figura 4.2. Modificación del contraste.....	42
Figura 4.3 Ecualización del Histograma (algoritmo clásico).....	43
Figura 4.4 Pruebas Sistema RCN. (a) Ecualización de la imagen (b) Detección de bordes algoritmo Sobel.....	44
Figura 4.5. Umbralización: (a) Otsu, IJ-Isodata & Intermodal;(b) media.....	44
Figura 4.6. Operaciones Morfológicas: (a) con umbralización Otsu y (c) con umbralización filtro.....	45
Figura 4.7 Filtros de detección de bordes (a) Imagen original (b) Sobel (c) Canny.....	45
Figura 4.8 Umbralización metodología aplicación (a) algoritmo Isoda (b) Otsu (c) SysThreshold (d) algoritmo clásico selección automática del umbral.....	46
Figura 4.9 Algoritmo iterativo para la selección del umbral.....	46
Figura 4.10. Extracción de regiones: (a) sin utilizar parámetro tamaño; (b) utilizando parámetro tamaño.....	47
Figura 4.11. Elipse encontrada en el sistema RCN 0°.....	47
Figura 4.12. Imágenes seleccionadas (a) N. Ceranae (b) N. Ceranae y N. Apis.....	48
Figura 4.13. Resultados prueba efectividad imagen N1 (a) sistema RCN (b) plugin RCN	48
Figura 4.14. Resultados prueba efectividad imagen I2(a) imagen original N-SGE8; (b) plugin RCN; (c) sistema RCN	50

INDICE DE TABLAS

Tabla 2.1. Diferencias herramientas y librerías de procesamiento.....	20
Tabla 3.1. Resultados de las propiedades estadísticas de los histogramas de las imágenes del conjunto de muestreo.....	34
Tabla 3.2. Resultados del circularidad para las esporas con mayor valor de largo y ancho.....	38
Tabla 4.1. Resultados de clasificación imagen I1.....	58
Tabla 4.2. Resultados de clasificación imagen I2.....	58

1. Introducción

1.1 Motivación

En la última década el agente patológico *Nosema Ceranae* (N. Ceranae) ha provocado una imperceptible pandemia, considerada aún una enfermedad emergente. En comparación con *Nosema Apis* (N. Apis), N. Ceranae ha sido considerada como un importante problema de salud, tanto en las abejas y particulares, como en la colonia entera (Higes, Martín & Meana, 2010), ocasionando pérdidas significativas en la producción de la miel y en la eficiencia de la polinización.

Es este último el origen de la motivación principal para desarrollar mi proyecto fin de máster, debido a que el proceso de polinización juega un papel importante en la regeneración natural del bosque. Si este proceso no se realiza, muchas especies y procesos del ecosistema conectados entre sí, podrían dejar de existir.

En consecuencia, si el diagnóstico que se realiza durante el control de enfermedades infecciosas en los animales productores de alimentos, como lo son las abejas falla, no se tomarán las medidas adecuadas para tratar las abejas y colmenas enfermas, ocasionando así la propagación de la enfermedad mortal.

Debido a esto el proyecto propone un sistema de reconocimiento y clasificación de los microsporidio de *N. Apis* y *N. Ceranae*, para servir de apoyo al diagnóstico de la enfermedad parasitaria Nosemosis de tipo C.

Objetivos Generales

1. El objetivo de este trabajo es automatizar el proceso de detección y clasificación de esporas, considerando y evaluando las diversas metodologías de tratamiento de imágenes y reconocimiento de patrones que existe en el campo de la visión artificial, relacionada con el ámbito de Visión por Computador, con el fin de seleccionar una metodología o crear una nueva, combinando diversas técnicas que permitan realizar el proceso de clasificación de esporas y diagnostico automático mediante un sistema fácil, rápido y fiable.
2. Desarrollar un sistema de gestión de datos, que sea capaz de almacenar, procesar y validar la ingente cantidad de datos que aportará el nuevo sistema analítico, proporcionando una serie de alarmas que alerten sobre las anomalías que puedan estar presentes en la muestra analizada, utilizando una serie de reglas de decisión predefinidas.
3. Automatizar el laboratorio de diagnóstico de parasitología del hospital clínico veterinario de la facultad de Veterinaria de la Universidad Complutense de Madrid, desarrollando herramientas hechas a la medida de sus necesidades (utilizando la informática como apoyo) y metodologías organizativas (e.g. flujo de trabajo) que mejoren los resultados y obtengan como respuesta la satisfacción de los usuarios del laboratorio. Además de estudiar la posibilidad de obtener la certificación por la norma ISO-9001 o, más recientemente, de acreditación por la norma ISO-15189.

Objetivos Específicos

- Analizar las propiedades del conjunto de imágenes tomadas en el laboratorio, para determinar si el sistema de captura e iluminación pueden ser procesados.
- Estudiar y analizar los algoritmos de reconocimiento de formas, que sirvan para detectar objetos elipsoidales y reconocer sus dimensiones.

- Comparar los resultados de las metodologías estudiadas aplicándolas a un mismo conjunto de imágenes, para evaluar su desempeño.
- Analizar los métodos de diagnóstico de la Nosemosis de tipo C.
- Asegurar la formación técnica de los investigadores.

1.2 Estructura del trabajo

En el capítulo 2 se realiza la descripción detallada del problema del despoblamiento de colmenas, su origen, nuevos estudios e investigaciones y los métodos diagnósticos existentes, además se exponen los diferentes trabajos que se han realizado y los algoritmos que se utilizan para realizar el tratamiento de las imágenes, ordenados desde los procedimientos más generales (filtros), hasta los más específicos (detección de elipses).

En el capítulo 3 se detallan las dos metodologías desarrolladas, que abarcan desde el procesamiento de las imágenes, reconocimiento de formas y cálculo de dimensiones hasta la clasificación de los microsporidios.

Las comparaciones de resultados de los dos sistemas se entregan en el capítulo 4, discutiendo y comparando los resultados de clasificación y de reconocimiento de esporas en el conjunto de imágenes muestreadas.

La memoria termina con el capítulo 5, presentando las conclusiones y trabajos futuros.

2. Estado del Arte

2.1 El problema del despoblamiento de colmenas

Síndrome de despoblamiento de las Colmenas (SDC)

El síndrome del despoblamiento se caracteriza por la disminución del número de abejas de una colonia, sin causa aparente, hasta que las abejas sobrevivientes no pueden mantener las tareas elementales de la colonia, provocando que entre en colapso y desaparezca sin presentar una sintomatología que permita intuir el origen del problema (García, Meana, Higes, Martín, Sanz, A.; & Sanz, 2005)



Figura 2.1. Diagnostico en colmenas. Una colmena sana alberga millares de abejas (izquierda), pero si sufre el síndrome del despoblamiento (derecha), acaba perdiéndolas y muere (fuente: Meana, Martín & Higes, 2009).

El denominado *Síndrome de despoblamiento de colmenas* (ver figura 2.1), no es un fenómeno nuevo. Se detectó en España, a finales de los 90's y principios del 2000 aunque con una difusión inferior a la actual. Existen diversas posibles causas del despoblamiento, que se investigaron hasta encontrar la causa real, que varían desde una hipótesis de despoblamiento a causa del tratamiento de las semillas de girasol con insecticidas hasta enfermedades parasitarias de las abejas (Meana, Martín & Higes, 2009). Será esta última la causa real y el objeto de nuestro interés.

Detección de la enfermedad mortal "Nosemosis C"

La elevada mortalidad de las colonias de abejas denunciadas por los apicultores y veterinarios durante el periodo 2000/2005, hizo sospechar que podría ser causa de enfermedades parasitarias, debido a que se experimentó un crecimiento exponencial en los casos diagnosticados como nosemosis (OIE, 2008), durante ese periodo.

La Nosemosis es una enfermedad parasitaria causada por un microsporidio, *Nosema Apis*, que se localiza y desarrolla en el interior de las células epiteliales del ventrículo, es decir, en el intestino medio de las abejas adultas.

Las lesiones macroscópicas y la detección de esporos con morfología compatible con los de *N. Apis* (ver figura 2.2), hizo pensar que la enfermedad parasitaria era nosemosis. Sin embargo, el cuadro clínico no coincidía totalmente con la descripción de la enfermedad en las publicaciones especializadas y las características moleculares presentaban pequeñas diferencias en sus dimensiones. El tamaño aproximado del microsporidio *N. Apis* es $4,5-7,0 \times 2,5-4,0 \mu\text{m}$ y las dimensiones de las esporas detectadas eran aproximadamente de $3,35-4,1 \times 1,4-1,9 \mu\text{m}$.

Se había detectado un agente potencialmente patógeno (*N. Apis*), que no era el causante de esta enfermedad mortal, pero que tenía una relación estadística entre sus esporas y la desaparición de las abejas.

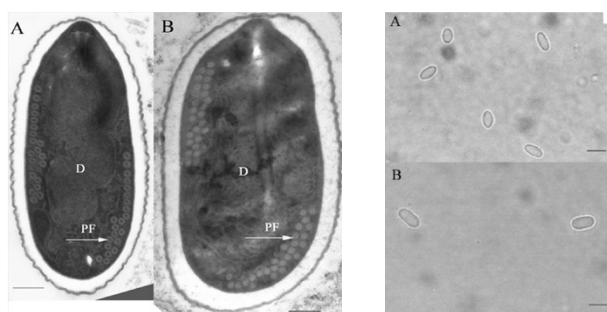


Figura 2.2. *N. Ceranae* (a) y *N. Apis* (b)

Se realizó un cambio en la metodología al investigar un nuevo parásito: *N. Ceranae*, que afectaba hasta el momento solamente a las abejas asiáticas. Gracias a este cambio se detectó que este nuevo agente patógeno es el causante de la enfermedad mortal, conocida ahora como Nosemosis de tipo C (Coloss, 2009).

Comenzando así una carrera a contrarreloj en donde se buscan, entre otras cosas, nuevos métodos diagnósticos para detectar el parásito. Siendo este nuestro objetivo principal.

Control de las enfermedades parasitarias

El control de las enfermedades infecciosas en los animales que producen alimentos es un aspecto esencial de la medicina veterinaria e incluye el diagnóstico de la enfermedad así como las medidas para tratar los animales enfermos y evitar la propagación de la enfermedad. El control de las enfermedades de las abejas productoras de miel es necesario para mantener su papel como productor de alimentos (miel, polen, etc) y como polinizadores de los cultivos y vegetación silvestre (Higes, Martín & Meana, 2010).

Métodos de Diagnóstico

Los métodos tradicionales utilizados en la detección de *N. Ceranae* se basan en análisis microscópicos (OIE, 2008) (Fig.2.3), métodos moleculares o microscopia electrónica de transmisión y técnicas de PCR (Higes, Martín & Meana, 2010).

Ahora es más común utilizar sistemas de captura de datos, que poseen herramientas robustas o módulos de procesamiento de imágenes que permiten realizar el tratamiento y reconocimiento completo de las partículas o microorganismos en las imágenes muestreadas.

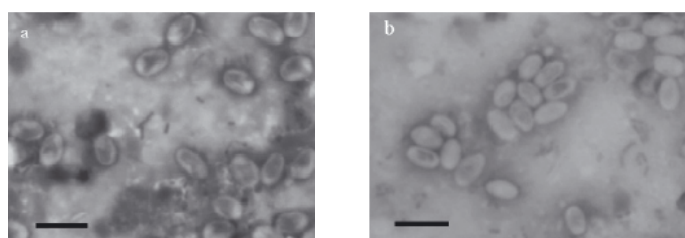


Figura 2.3. Imágenes microscópicas esporas de *N. Apis* (a) y *N. Ceranae* (b)

Desde la época de los 90's se han desarrollado e implantado sistemas informáticos en los laboratorios (Furundarena, 2009) (Anguita & Pérez, 2009), con el fin de clasificar y/o diagnosticar enfermedades a través de visión artificial.

El diagnóstico de enfermedades parasitarias en las abejas se sigue realizando mediante los métodos tradicionales de análisis microscópicos. Sin embargo, las investigaciones desarrolladas en la fase de detección del ciclo biológico de los microsporidios *N. Apis* y *Ceranae* (Martín, Meana, García-Palencia, Marín, Botías, Garrido-Bailón, Barrios & Higes, 2009), utilizan técnicas de procesamiento de imágenes para observar los cambios de temperatura para analizar si el ciclo de vida depende de la temperatura (ver figura 2.4).

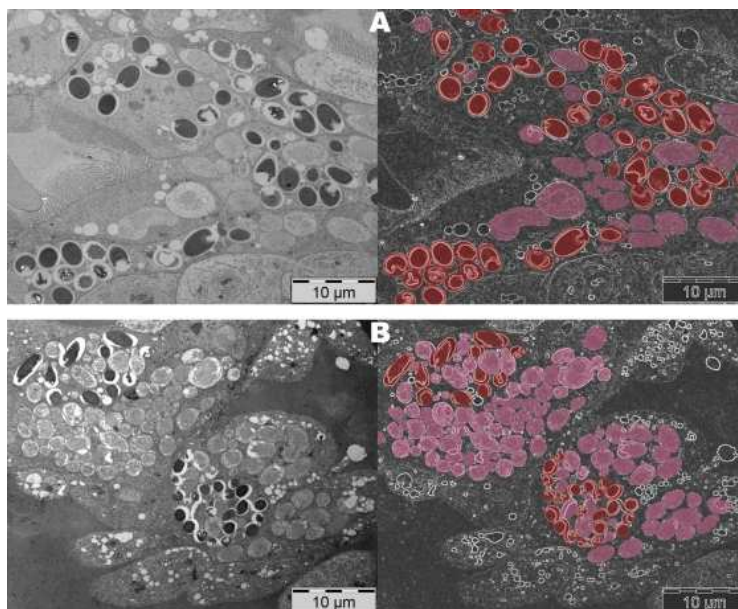


Figura 2.4. Vistas detalladas de las células epiteliales del ventrículo parasitados a los 7 días a una temperatura de 33 ° C. Las células de *N. Apis* infectadas (A) muestra cantidades similares de estados inmaduros y maduros (rojo), mientras que las células de *N. Ceranae* infectadas (B) mostraron una mayor proporción de formas inmaduras (color rosa) en ese momento. (fuente: Martín, Meana, García-Palencia, Marín, Botías, Garrido-Bailón, Barrios & Higes, 2009)

Para otro tipo de microsporidio como lo es N. Bombi se utilizaron técnicas de procesamiento de imágenes (medición), para calcular la intensidad de la infección producida por este parásito (Otti & Schmid-Hempel, 2007).

Estos son algunos de los trabajos más conocidos desarrollados con técnicas de visión por computadora para analizar y medir microsporidios.

Otros trabajos que han utilizado técnicas de visión por computadora para clasificar, medir y cuantificar partículas parecidas a las esporas, citan la problemática de detección, caracterización (forma, tamaño y color), en (Sossa, Osorio, Prieto & Angulo, 2002) para realizar la clasificación de núcleos de células en las imágenes de microscopio de citologías del epitelio escamoso. Otras estrategias de gestión requieren la cuantificación y medición del tamaño y distribución de partículas de polvo (Igathinathane, Melin, Sokhansanj, Bi, Lim, Pordesimo & Columbus, 2009), entre otros.

2.2 Algoritmos de procesamiento de imágenes

Los algoritmos que se analizan en esta sección sirvieron como referencia para aplicar secuencias y/o combinaciones en las metodologías desarrolladas para reconocer, medir y clasificar los microsporidios de las imágenes. Todos ellos fueron probados en dos herramientas de procesamiento de imágenes (*ImageJ* e *Image Processing Lab* (AForge.Net)), y algunos de ellos que no existían en las herramientas, fueron implementados en c# y java para comprobar su funcionamiento y efectividad.

2.2.1 Eliminación de Ruido

Todas las imágenes tienen una cierta cantidad de ruido, bien sea por el medio de transmisión de la señal o por el sensor de la cámara. La manifestación del ruido ocurre generalmente en píxeles aislados que toman un valor de gris diferente al de sus vecinos. Para saber qué tipo de filtro se debe aplicar, primero hay que conocer las características del conjunto total de imágenes respecto al ruido, con el fin de clasificar su tipo (De la Escalera, 2001).

El ruido se puede clasificar en 4 tipos: Gausiano, Impulsional, Frecuencial y Multiplicativo, donde el tipo de ruido que más se asemeja al de las imágenes es el Gausiano, debido a que éste produce pequeñas variaciones en la imagen, ya sea por la ganancia del sensor o ruido generado en el proceso de digitalización, entre otros.

Filtro Gaussiano

Si se considera una imagen como una función real de dos variables $I(x, y)$, este tipo de filtro es una convolución entre la señal de entrada (la imagen $I(x, y)$) y otra función real $g(x, y)$ (una función gaussiana bidimensional por ejemplo). En términos genéricos la aplicación de una convolución a una imagen se puede escribir discretamente como:

$$g(i, j) * I(i, j) = \sum_{k=1}^m \sum_{l=1}^n g(k, l) I(i - k, j - l) \quad (2.1)$$

En la ecuación anterior m y n son el ancho y largo de la imagen en píxeles. Es importante resaltar que por la simetría de la función es posible escribir el filtro bidimensional como dos filtros unidimensionales, permitiendo implementaciones rápidas ($O(n)$ para el número de píxeles) y simples (Cerdeira, 2007).

$$\sum_{k=1}^m e^{-\left(\frac{k^2+l^2}{\sigma^2}\right)} * f(i - k, j - l) = \sum_{k=1}^m e^{-\left(\frac{k^2}{\sigma^2}\right)} \left(\sum_{l=1}^n e^{-\left(\frac{l^2}{\sigma^2}\right)} I(i - k, j - l) \right) \quad (2.2)$$

Donde σ es el ancho de la campana gaussiana. La manera más simple de ver el efecto de usar el filtro gaussiano anterior es considerar su versión bidimensional. Cada nuevo pixel queda definido por el promedio de los valores de sus píxeles vecinos, dándoles más peso mientras estén mas cerca al pixel que se está procesando (Nixon & Aguado, 2005). Esto requiere de un parámetro (σ) que corresponde, al ancho de la campana gaussiana.

2.2.2 Conversión a escala de gris

Las imágenes se convierten a escala de grises para poder manipularlas, donde los datos de la imagen consisten en un solo canal que representa la intensidad, brillo o densidad de la imagen. En la mayoría de los casos, los valores positivos son los que tienen sentido (e.g la intensidad de la luz). Típicamente una imagen en escala de gris usa 8 bits (1 byte) por pixel y el rango de los valores de intensidad se encuentra entre [0-255], donde el valor mínimo del rango representa el brillo mínimo (Negro) y el valor máximo representa el brillo máximo.

2.2.3 Transformación Lineal

Para ajustar el contraste general de la imagen en escala de grises se multiplicó una constante de parcialidad c (*bias*) al valor de cada uno de los píxeles de la imagen (Ecuación 2.3). La constante de parcialidad cumple la condición $1 < c \leq 2$, con el fin de aumentar el contraste ($1 < c$), sin llegar al extremo del brillo $c \leq 2$.

Los valores máximo y mínimo de la constante de parcialidad se obtuvieron realizando pruebas desde un valor mínimo, que cumplía con la condición básica de aumento de contraste, donde $c > 1$, y fue aumentando progresivamente hasta su valor máximo ($c = 2$), creando para el algoritmo una nueva condición máxima del aumento del contraste $c \leq 2$.

El brillo fue ajustado de forma similar (Ecuación 2.3), adicionando una constante de ganancia $b = 0.5$ al valor de cada uno de los píxeles de la imagen. El valor de la constante b sirve para redondear al entero más cercano antes que se trunque el nuevo valor al convertir el resultado a un tipo entero (casting).

$$g_{cte}(x, y) = f(x, y) * c \quad (2.3)$$

$$g_{brillo}(x, y) = c * f(x, y) + b \quad (2.4)$$

2.2.4 Ecualización del histograma

A través de la ecualización del histograma se busca obtener las características más definidas en las imágenes, como lo son las esporas. Esta técnica transforma o modifica el histograma de una imagen con el fin de realzar su contraste expandiendo la distribución de los niveles de gris de forma uniforme (Pajares & De la Cruz, 2001).

Un histograma ideal sería plano, con el mismo número de píxeles en cada nivel de gris (ver figura 2.5). Sin embargo, no hay forma de eliminar o disminuir los diferentes picos individualmente en un histograma, por lo tanto una verdadera distribución uniforme es imposible de alcanzar.

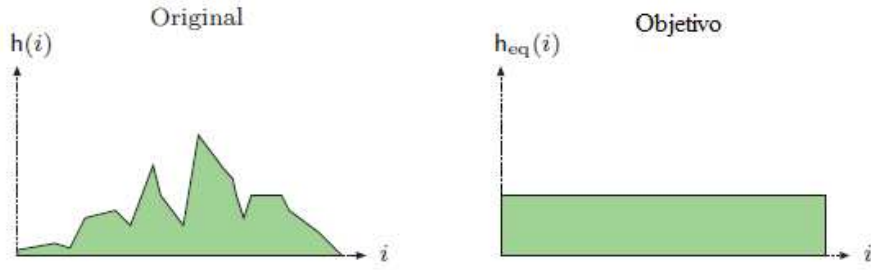


Figura 2.5. Histograma original - histograma ideal

Pero si se utiliza como ideal un histograma acumulativo (Burge & Burge, 2009), la distribución uniforme de la imagen será representada en el histograma como una rampa lineal, como se muestra en la figura 2.6.

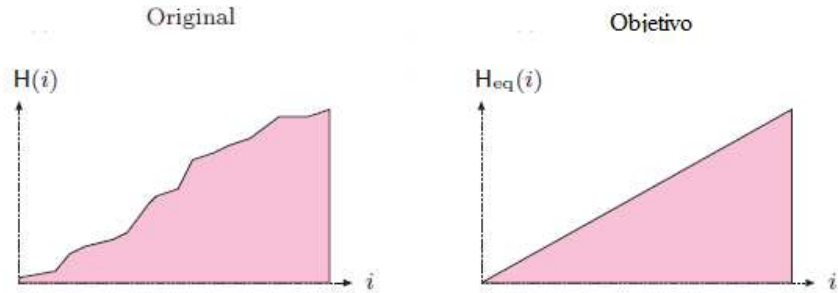


Figura 2.6. Histograma Original- Histograma acumulativo

De esta forma se puede replantear nuevamente el objetivo, encontrando una operación de punto tal que permita cambiar las líneas del histograma para que sea aproximadamente lineal.

La operación de punto (ecuación 2.3) deseada se obtiene del histograma acumulativo (H) de la imagen original, de tamaño $m \times n$, su rango de intensidad es igual a $[0, k-1]$ y los píxeles (p) de la imagen se encuentran entre el rango.

$$f_{eq}(p) = \left\lceil H(p) * \frac{k-1}{m*n} \right\rceil \quad (2.3)$$

El resultado de la función que se obtendrá será monótonamente creciente, porque $H(p)$ es monótona y k, m y n son constantes positivas.

2.2.5 Algoritmos de Segmentación

Al realizar el proceso de segmentación se buscó dividir la imagen en los objetos que la conforman, realizando subdivisiones al detectar los objetos de interés (e.g. esporas). La mayoría de los algoritmos de segmentación se basan en dos propiedades básicas de los niveles de gris de la imagen: similitud y discontinuidad. La primera categoría se basa en técnicas de umbrales (división y fusión), mientras que en la segunda categoría se divide la imagen basándose en cambios bruscos en el nivel de gris (e.g. bordes) (Pajares & De la Cruz, 2001).

Algoritmos de similitud: Técnicas de umbralización

Umbralización (Thresholding). Las imágenes como hemos visto, están compuestas de microorganismos (e.g. esporas) oscuros sobre un fondo claro, de tal forma que los niveles de gris están agrupados en dos tonalidades diferenciadas. Una de las formas de separar los microorganismos del fondo consiste en seleccionar un umbral T que se encargue de separar dichos nodos.

Para clasificar cada uno de los píxeles $f(x, y)$ se evalúa la siguiente regla:

$$f_t(x, y) = \begin{cases} 0 & , f(x, y) < T \\ 1 & , f(x, y) \geq T \end{cases} \quad (2.4)$$

Si el pixel cumple con la primera condición, se le etiqueta como fondo ($f(x, y) = 0$); de lo contrario, como microorganismo ($f(x, y) = 1$). De esta forma se clasifican los píxeles de la imagen en dos categorías [0-1] (ver figura 2.7).

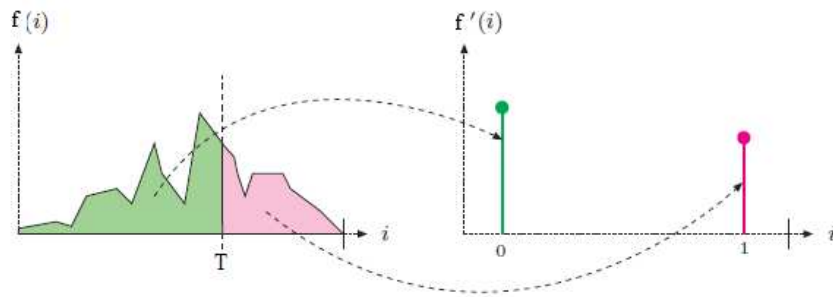


Figura 2.7. Clasificación de dos categorías del histograma

Isodata (Iterative Self-Organizing Data Analysis Techniques). El algoritmo isodata (*Iterative Self-Organizing Data Analysis Techniques*) (Trier & Jain, 1995) (Yang & Yan, 2000), es uno de los algoritmos más sofisticados que permite ajustar automáticamente el número de grupos durante la iteración, mediante la fusión y agrupación de grupos con grandes desviaciones estándar (Venkateswarlu, & Raju, 2003).

El método divide la imagen en grupos y fondos tomados del umbral inicial; a continuación se calculan los promedios de los píxeles menores o iguales al umbral y mayores que el mismo. Los promedios de los dos valores se calculan, el umbral se incrementa (Ecuación 2.4) y el proceso se repite hasta que el umbral es mayor que el promedio compuesto.

$$Umbral = (promedio\ fondos + promedio\ grupos)/2 \quad (2.4)$$

Algoritmos de discontinuidad: Sobel y Canny

Algoritmo de detección de bordes Canny. El algoritmo Canny (Canny, 1986), está considerado como uno de los mejores métodos de detección de contornos mediante el empleo de máscaras de convolución y basado en la primera derivada, debido a que toma el valor de cero en todas las regiones donde no varía la intensidad y tiene un valor constante en toda la transición de intensidad. Por tanto un cambio de intensidad se manifiesta como un cambio brusco en la primera derivada, característica que es usada para detectar un borde, y en la que se basa el algoritmo de Canny.

Se divide en tres pasos:

- a. Obtención del gradiente (magnitud y ángulo en cada pixel)
- b. Supresión no máxima: Adelgazamiento del ancho de los bordes obtenidos por el gradiente, hasta lograr bordes de un píxel de ancho.
- c. Histéresis de umbral al resultado de la supresión no máxima.

Resulta particularmente interesante porque extrae bordes y cierra los contornos evitando posibles rupturas durante su extracción (Pajares & De la Cruz, 2001). Esta es una de las características que más nos interesa al obtener los bordes de la espora y su membrana.

Algoritmo de detección de bordes Sobel, El operador Sobel es uno de los detectores de bordes que más se utiliza en las investigaciones realizadas hoy en día. Es un operador de gradiente que magnifica el ruido subyacente en la imagen, minimiza la aparición del ruido y tiene la propiedad de suavizar la imagen, eliminando parte del ruido.

2.2.6 Detección de Elipse

La detección de elipse es considerada como una de las tareas más importante en el reconocimiento de patrones. Durante las dos últimas décadas los investigadores han desarrollado diversos enfoques para la detección de elipses, los cuales pueden ser divididos en dos grupos: clustering (Yin, 1998) (Dave, 1992) y optimización (Fitzgibbon, 1999), (Zhang, S.C. & Liu, Z.Q, 2004).

Para nuestro desarrollo aplicaremos un algoritmo de optimización de detección de elipses (McLaughlin, 1998), que integra las propiedades de direccionalidad de bordes para descomponer los parámetros de la transformación de Hough y hacer un uso completo de los puntos candidatos para conseguir la eficiencia computacional.

El algoritmo consiste en tres pasos:

1. Extracción de bordes usando el algoritmo de detección Sobel; después se realiza un proceso para remover los puntos que no son parte del borde de la elipse.
2. Extracción de regiones de interés (elipses).
3. Se calculan los parámetros de la elipse de la región (Algoritmo para calcular los parámetros de la elipse de una región).
4. Se seleccionan las elipses validas.

Algoritmo para calcular los parámetros de la elipse de una región

Una elipse puede ser descrita de dos maneras; a través de su coordenada céntrica, radio mayor, radio menor y ángulo de orientación (p, q, a, b, θ) o a través de su coordenada céntrica, radio fuera del foco 1, radio afuera del foco 2 y el ángulo de orientación (p, q, a, b, θ) , nosotros usaremos la primera descripción, puesto que las variables $(p, q, a$ y $b)$ se obtienen de la ecuación derivada para encontrar la elipse en la imagen (Ecuación 2.5).

$$f(\vec{a}, \vec{x}) = a(x - p)^2 + 2b(x - p)(y - q) + c(y - q)^2 = 1 \quad (2.5)$$

Existen seis pasos para determinar los parámetros de la elipse de una imagen, comenzando por encontrar las coordenadas del centro de la elipse para determinar la longitud del radio mayor (a), la longitud del radio menor (b) y la distancia media entre el foco (c). Estos seis pasos serán descritos detalladamente a continuación:

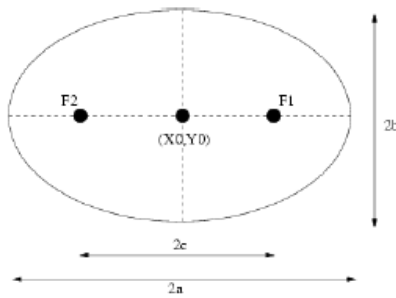


Figura 2.8. Anatomía de la elipse

1. Determinar el centro de la elipse

1.1 Seleccionar los tres puntos X_1 , X_2 y X_3

Tres puntos son seleccionados aleatoriamente de la imagen de tal manera que cada punto tiene la misma oportunidad de ser elegido. La imagen se divide en 4 secciones, en tres de las cuatro secciones se selecciona un punto aleatorio de su borde por sección. En la figura 2.9, se muestran los tres puntos ideales.

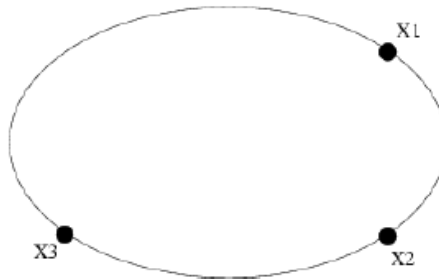


Figura 2.9. Selección de tres puntos aleatorios en 3 de las 4 secciones de la elipse

1.2 Se determina la ecuación de la recta para cada punto donde la pendiente de la recta es el gradiente en el punto $y = mx + b$. Esto se realiza evaluando los píxeles alrededor del punto y calculando un ajuste de rectas de mínimos cuadrados entre ellos.

1.3 Determinar la intersección de las tangentes que pasan a través de los puntos (x_1, x_2) y (x_2, x_3) . Los puntos de intersección de la tangente t_{12} y t_{23} (ver figura 2.10), se encuentran solucionando el sistema de ecuaciones lineales para las coordenadas (x, y) . (ecuación 2.6 y 2.7)

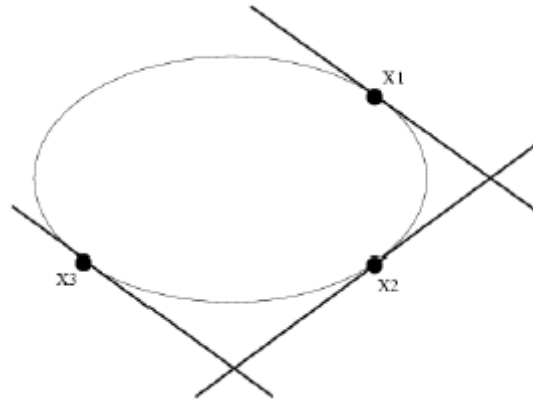


Figura 2.10 Rectas tangentes puntos aleatorios seleccionados

Tangente X_1 y X_2 para t_{12} :

$$\begin{bmatrix} m_1x + b_1 - y = 0 \\ m_2x + b_2 - y = 0 \end{bmatrix} \quad (2.6)$$

Tangente X_2 y X_3 para t_{23} :

$$\begin{bmatrix} m_2x + b_2 - y = 0 \\ m_3x + b_3 - y = 0 \end{bmatrix} \quad (2.7)$$

1.4 Calcular la bisectriz de los puntos de intersección tangencial. Esta es una línea de intersección de la tangente, t , hasta el punto medio de los dos puntos. Ver figura 2.11.

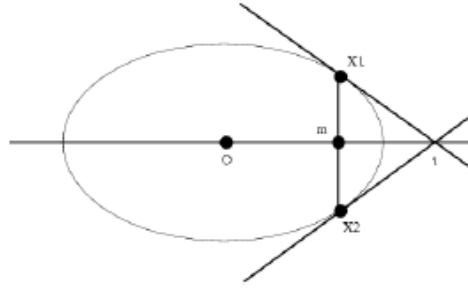


Figura 2.11. Bisectriz de los puntos de intersección tangencial.

Las coordenadas del punto medio m_{12} son iguales a la distancia media de X_1 a X_2 . Las coordenadas del punto medio y las coordenadas de la intersección t_{12} son usadas para obtener la ecuación lineal de la intersección. Al solucionar la ecuación 2.8, se encuentra la ecuación lineal de la intersección.

$$pendiente = \frac{m_y - t_y}{m_x - t_x} \quad (2.8)$$

Y usando la pendiente en la ecuación de la recta para encontrar la intersección en el eje y:

$$b = pendiente * x - y = pendiente * t_x - t_y \quad (2.9)$$

La línea de intersección es igual a:

$$y = pendiente * x - b \quad (2.10)$$

1.5 Las coordenadas de la intersección son encontradas usando las ecuaciones de la recta de las intersecciones determinada en el paso anterior en el siguiente sistema de ecuaciones lineales:

$$\begin{cases} m_1x + b_1 - y = 0 \\ m_3x + b_3 - y = 0 \end{cases} \quad (2.11)$$

1.6 Después de obtener la coordenada céntrica de la elipse (x_0, y_0) , se reduce la ecuación 2.5 a $(ay^2 + 2bxy + cy^2 = 1)$ usando los tres puntos aleatoriamente seleccionados y a partir de esta nueva ecuación se crean las ecuaciones lineales con respecto a los parámetros a, b y c, para los tres parámetros seleccionados.

$$\begin{cases} ax_1^2 + 2bx_1y_1 + cy_1^2 = 1 \\ ax_2^2 + 2bx_2y_2 + cy_2^2 = 1 \\ ax_3^2 + 2bx_3y_3 + cy_3^2 = 1 \end{cases} \quad (2.12)$$

Solucionando este sistemas de ecuaciones se obtiene el radio mayor (a) y el radio menor (b).

2.2.7 Reconocimiento de patrones

Cuando se habla de reconocimiento de patrones en imágenes, generalmente se refiere al problema de encontrar cierta forma o figura dada con una cierta tolerancia dentro de una imagen. Como sabemos, este problema puede no tener una solución y menos aún se puede asegurar que de existir una, esta sea única. Basta pensar como analogía, que una persona muchas veces no está realmente segura de lo que está viendo, ya que puede confundir o ver objetos que no está viendo realmente. Sin embargo, a pesar de esta falta aparente de precisión, una persona muchas veces es capaz de dar una excelente respuesta cuando se trata de reconocer una figura en una imagen.

Una de las preguntas que surge de esta situación es si una imagen contiene o no suficiente información para entenderla completamente, o es el observador, que con sus experiencias y conocimiento previo, logra describir correctamente las imágenes. Por ello la importancia que se le ha dado a la necesidad de incorporar información adicional acerca de las imágenes para un buen procesamiento.

Los rangos de dimensiones de los microsporidios *N. Apis* y *N. Ceranae* (sección 2.1), nos sirve de información adicional para incorporar a la solución del problema de reconocimiento y clasificación de esporas. gracias a que la clasificación es específica entre esos tipos de microsporidios.

2.2.8 Herramientas y librerías de procesamiento de imágenes

ImageJ

Es un software de procesamiento de imágenes, de dominio público (open source), implementado completamente en Java y por lo tanto es multi-plataforma. Ofrece un conjunto

de herramientas prefabricadas para la manipulación interactiva de las imágenes y puede extender su funcionalidad fácilmente a través de componentes de software (Plugins).

ImageJ fue diseñado y desarrollado por Wayne Rasband (Rasband, 2009) en el Instituto Nacional de Salud de EE.UU (NIH), con una arquitectura abierta que provee extensibilidad a través de sus Plugins, que son pequeños módulos o clases independientes que pueden ser compilados, cargados y ejecutados en tiempo ejecución. (Ver figura 2.9).

Una larga lista de plugins (e.g. Colección MBF, disponibles en <http://rsb.info.nih.gov/ij/plugins/mbf-collection.html>), refleja el uso del software ImageJ a lo largo de una serie de campos en la ciencia y la ingeniería.

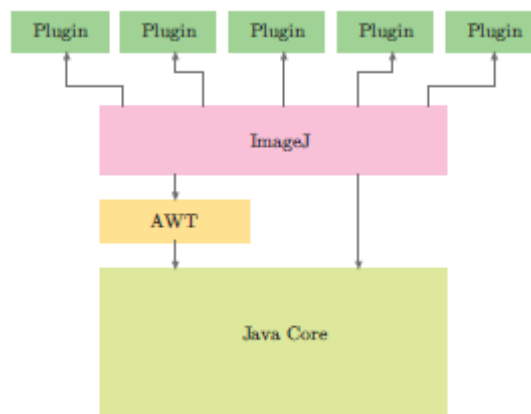


Figura 2.12. Estructura simplificada arquitectura del software ImageJ

Sin ser exhaustivos, algunos ejemplos en los que se ha utilizado uno o más de estos plugins serán citados a continuación.

1. El método para contar eritrocitos es presentado en (Gering & Atkinson, 2004), donde los autores describen un protocolo para la generación rápida del conteo de eritrocitos a través del análisis digital de la imagen del frotis sanguíneo, con el fin de estimar la intensidad de la infección del parásito hematozoon en el huésped.
2. Estudios histopatológicos (Girish & Vijayalakshmi, 2004), se realiza una comparación y evaluación de la evolución del software (NIH Image / ImageJ) aplicando técnicas de conteo de células, herramientas de umbralización para segmentar las imágenes, entre otros procedimientos. Entre otros,

Gracias a esta característica se ha convertido en una plataforma ideal para el desarrollo y testeo de nuevas técnicas de procesamiento y algoritmos. Utilizada regularmente en investigación y desarrollo de aplicaciones en laboratorios de todo el mundo (Collins, 2007), sobre todo en biología y medicina.

IpLab (aForge.net)

AForge (Kirillov, 2007) es un framework de dominio público (open source c#), diseñado para desarrolladores e investigadores en el campo de visión artificial, relacionada con los ámbitos de la Visión por Computador e Inteligencia Artificial y Procesamiento de imágenes, entre otros. Ofrece una herramienta de procesamiento prefabricada para la manipulación interactiva de las imágenes, *Imaging Processing Laboratory* (ipLab), y un conjunto de librerías para hacer uso de su gran variedad de algoritmos.

Para nuestro proyecto se seleccionó la librería *AForge.Imaging*, con rutinas y filtros de procesamiento de imágenes, desarrollando el sistema en la plataforma Visual Studio .Net, bajo el lenguaje C#.

Es un framework que está mejorando continuamente, por lo que nuevas características y desarrollos están constantemente actualizándose.

<i>Ventajas</i>	<i>Desventajas</i>
Open Source	No esta desarrollado por una entidad distinguida
Funciones para el procesamiento de imágenes (algoritmos disponibles).	No está del todo centrado en la visión por ordenador (tratamiento de imágenes).
Plataforma de desarrollo requerido (Visual Studio) es robusta y de libre acceso.	No es multiplataforma, solo tiene soporte para Windows

Tabla 2.1. Diferencias herramientas y librerías de procesamiento

3. Metodología de análisis, clasificación y diagnóstico

Para obtener la mejor solución al problema enunciado en la sección 2.1, decidimos realizar dos aplicaciones con diferentes metodologías de tratamiento de imágenes, cuyo enfoque es totalmente diferentes.

El sistema RCN (Sistema de reconocimiento y clasificación de N. Apis, N. Ceranae), se desarrolló completamente en c#, utilizando las librerías *Aforge.net* y el *software Image Processing Lab* de dominio público, mientras que el plugin RCN se desarrolló en Java utilizando la plataforma de desarrollo y la herramienta de procesamiento de imágenes *ImageJ*, también de dominio público (ver sección 2.2.8).

En la figura 3.1, se muestra el diagrama general que se utilizó para procesar y analizar las imágenes tanto en el sistema como en el plugin, utilizando algunos de los algoritmos o combinaciones de ellos, evaluados en el capítulo anterior, con el fin de obtener los mejores resultados al reconocer y clasificar los microsporidios de las imágenes.

Cada una de las fases descritas en el diagrama serán expuestas detalladamente a lo largo del capítulo.

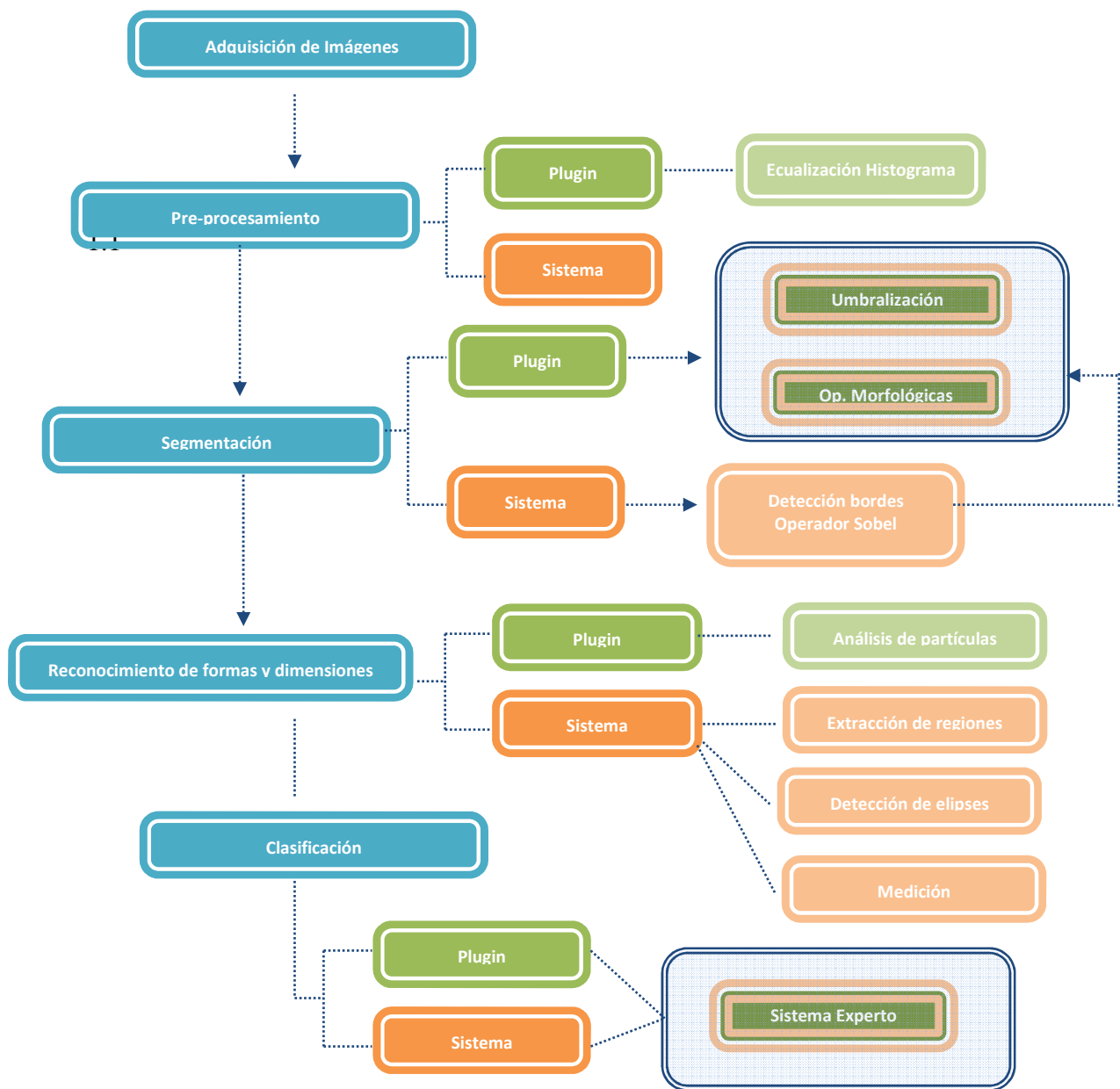


Figura 3.1. Diagrama global del tratamiento digital del conjunto total de imágenes

3.1 Adquisición de la Imagen

Las imágenes microscópicas se obtuvieron utilizando un sistema de captura compuesto por una cámara digital Nikon de 5.0 megapíxeles, acoplada a un microscopio de fluorescencia Nikon Eclipse E-400.

Se utilizó el software **NIS Elements 3.0**, para controlar el microscopio, la cámara y sus periféricos, además de interpretar, y digitalizar la imagen (Moreno & Marcos, 2006).

El microscopio posee un rango de sistema óptico, del cual se utilizaron los objetivos de 40x y 100x de aumento.

Para el sistema de iluminación se utilizó la opción predefinida para fotos del microscopio (Lámpara alógena 6V 30W).

El sistema de captura se encuentra en el hospital de la facultad de Veterinaria de la Universidad Complutense de Madrid, en las instalaciones del laboratorio de parasitología, en donde se tomaron tres sesiones de fotos de las muestras proporcionadas por el doctor Mariano Higes del centro agrario de Marchamalo.

Cada una de las sesiones será descrita a continuación:

Primera sesión: En la muestra del portaobjetos se añadió una coloración química, llamada Azul de Metileno (Ver figura 3.2.a), la cual se utilizó como colorante en la tinción para observar en el microscopio (mejorando el contraste de la imagen).

Las imágenes tomadas durante esta sesión fueron únicamente de las muestras de *N. Ceranae*. Total imágenes tomadas: 24.

Segunda sesión: Durante la segunda sesión también se tomaron imágenes de las muestras de *N. Ceranae*, sin coloración química (Ver figura 3.2.b) Total de imágenes tomadas: 31.

Tercera sesión: Durante la tercera y última sesión se tomaron fotos de las muestras de *N. Apis*, con coloración química (Ver figura 3.2.c). Lamentablemente en esta sesión no fue posible tener un conjunto de fotos con la cantidad mínima establecida (20 fotos), ya que las muestras de las esporas de *N. Apis* se encontraban en mal estado y solamente se pudieron sacar 3 fotos.

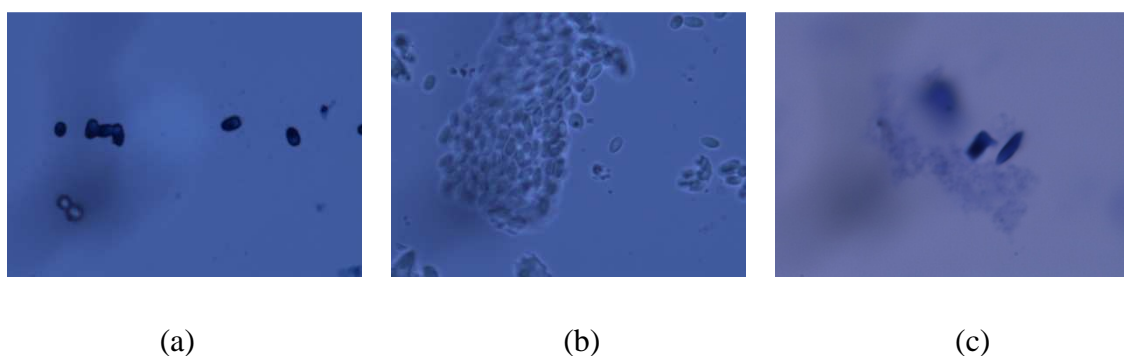


Figura 3.2. Imágenes digitales de las muestras (a) N. Ceranae con coloración (b) N. Ceranae sin coloración (c) N. Apis con coloración

Por cada toma se obtuvo una imagen a color digitalizada, de 640 x 480 pixeles, formato JPEG. El repositorio de imágenes tomadas se encuentra en el laboratorio de diagnóstico de parasitología del Hospital Clínico Veterinario de la facultad de Veterinaria.

3.2 Pre-procesamiento de la imagen

Plugin RCN. Para conocer los niveles de brillo y contraste del conjunto total de imágenes se seleccionó un conjunto de muestras del 10% de las imágenes del conjunto total. Su proceso de selección fue aleatorio (gracias a que las imágenes presentan características similares).

Para cada una de las imágenes en el conjunto de muestreo, se calculó la media y la varianza basándose en su histograma (Gonzalez & Woods, 1992), con el fin de conocer los niveles de brillo y contraste, respectivamente.

Imagen	Máximo	Mínimo	Media	Varianza	Moda
N1	161	62	122.38	2.94	126
N2	183	56	121.76	3.12	128
N5	150	66	118.12	2.66	122
N10	159	51	100.89	2.90	105
NL2	138	57	102.02	2.73	106
NSEG8	184	69	135.93	3.36	141

Tabla 3.1. Resultados de las propiedades estadísticas de los histogramas de las imágenes del conjunto de muestreo

Como podemos ver en los resultados en la tabla 3.1, la media del conjunto de muestreo se encuentra en un rango de intensidad entre [100.89-135.93]. Esto quiere decir que el brillo

general de las imágenes se encuentra en un nivel medio tendiendo a disminuir; los niveles de gris se encuentran concentrados hacia la parte media del histograma. Los resultados de la varianza están en un rango entre [2.73 – 3.36], esto nos indica que la distribución de los niveles de gris concentrados en la zona media del histograma presentan un contraste bajo (Ver figura 3.3).

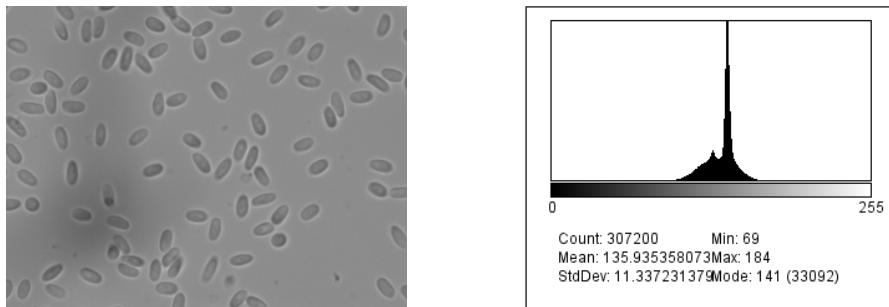


Figura 3.3. Imagen en escala de grises NSEG8 y su histograma

Para realzar el contraste de la imagen se utilizó el algoritmo de ecualización del histograma. (ver figura 3.4), el cual fue implementado en el lenguaje de programación *c#* y se utilizó el método *equalize()* de la plataforma de desarrollo de *ImageJ*.

```
private void igualacionHistograma ()
{
    int w = ip.getWidth();
    int h = ip.getHeight();
    int M = w * h;           // numero total de pixeles
    int K = 256;             // rango de intensidad [0-255]
    // Calculo del histograma acumulado
    int[] H = ip.getHistogram();
    for (int j = 1; j < H.length; j++) {
        H[j] = H[j-1] + H[j];
    }

    for (int v = 0; v < h; v++)
    {
        for (int u = 0; u < w; u++)
        {
            int a = ip.get(u, v);
            int b = H[a] * (K-1) / M;
            ip.set(u, v, b);
        }
    }
}
```

Figura 3.4. Implementación en código java de la ecualización del histograma lineal

3.3 Segmentación

Para simplificar el problema enunciado en la sección 2.1, decidimos atacarlo dividiendo el análisis de la imagen y reconocimiento de patrones en dos enfoques totalmente diferentes desde la etapa de segmentación, debido a que el éxito de la clasificación depende de la apropiada técnica de segmentación que se utilice (Masutani, Schiemann & Höhne, 1998) (Stefancik & Sonka, 2001).

El primer enfoque, divide la imagen basándose en cambios bruscos en el nivel de gris, como son los bordes. El segundo enfoque se basa en técnicas de umbralización (división y fusión) (ver sección 2.2.4).

Metodología Plugin RCN. Esta metodología sólo se basa en el segundo enfoque, utilizando la técnica de umbralización del filtro medio de las librerías *ImageJ*, para clasificar las regiones de interés (esporas) y binarizar las imágenes (Ver sección 4.2, plugin RCN).

Metodología Sistema RCN. Esta metodología combina los dos enfoques, utilizando el filtro Sobel de detección de bordes y la técnica de umbralización Otsu para binarizar las imágenes (Ver sección 4.2, Sistema RCN).

Operaciones Morfológicas (Plugin y Sistema RCN). Para simplificar la imagen obtenida del proceso de segmentación y preservar las formas principales de los objetos de interés, se utilizaron las operaciones morfológicas de erosión y dilatación (Pajares & De la Cruz, 2001). En la metodología del plugin se utilizó el orden: dilatación -> erosión, mientras que para la otra metodología se utilizó el orden inverso: erosión->dilatación. (Ver figura 3.5)

Como resultado se obtienen para ambas metodologías una imagen binaria, con las regiones clasificadas, en un rango de intensidad que sólo puede tomar dos valores [0-1], de la misma forma que sus píxeles, blanco o negro. Normalmente se codifican utilizando un solo bit (0/1) por píxel.

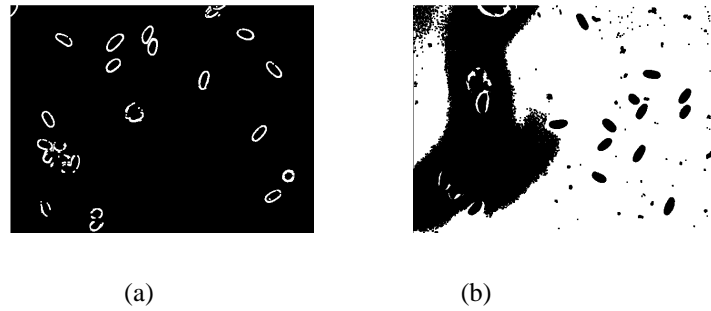


Figura 3.5. Operaciones Morfológicas: (a) Sistema RCN: dilatación -> erosión, (b) Plugin: erosión->dilatación

3.4 Reconocimiento de formas y dimensiones

En las dos metodologías se utilizaron los valores de los parámetros tamaño y circularidad para seleccionar los microsporidios que cumplen con un determinado número de píxeles.

Tamaño. Este parámetro se utiliza para excluir objetos que aparecen en la imagen binaria que no están en el rango de tamaño y que claramente no son objetos de interés. El tamaño mínimo se obtiene de las esporas de *N. Ceranae* (3,35-4,1 x 1,4-1,9 μm) y el máximo de las esporas de *N. Apis* (4,5-7,0 x 2,5-4,0 μm). Como se muestra a continuación:

$$\text{Tamaño Mínimo} = 33,5 \times 14 = 469 - \text{Tamaño Máximo} = 70 \times 40 = 2800$$

Circularidad. Para obtener el rango de circularidad de las imágenes, se calculó el ancho (a) y el largo (b), para cada una de las elipses de las imágenes del conjunto de muestreo, mediante la herramienta de procesamiento de imágenes *ImageJ*, con el fin de seleccionar dos elipses (la más larga y la más angosta) por cada imagen.

Al terminar la selección, se calculó la circularidad para la ó las elipses elegidas con la siguiente ecuación de circularidad.

$$C \approx \frac{2ab}{a^2+b^2} \quad (3.1)$$

Como podemos ver en la tabla 3.2, obtuvimos el rango de circularidad de las imágenes [0,69 - 1] en el conjunto de muestreo. Es importante resaltar que se obtuvo el mayor valor de circularidad en algunos microsporidios de la imagen N5 (aun siendo las esporas elipsoidales), en el caso de que estuvieran rotadas y perdieran su forma elipsoide.

Imagen	A	B	C
N1	41.08	18.16	0.74
	39.95	17.05	0.73
N2	40.19	19.39	0,79
	26.16	22.82	1
N5	29.83	20.70	0,94
	40.85	16.11	0,69
N10	39.38	16.59	0,72
	39.38	16.59	0,72
NL2	41.87	16.90	0,7
	41.87	16.90	0,7
NSEG8	43.72	18.79	0,73
	40.84	16.03	0,69

Tabla 3.2. Resultados del circularidad para las esporas con mayor valor de largo y ancho

Metodología Plugin RCN. Una vez la imagen ha sido segmentada, se realiza el análisis de partículas para obtener información sobre el tamaño y cantidad de las esporas. La implementación automática del analizador de partículas se llevó a cabo configurando los parámetros: tamaño y circularidad calculados anteriormente (ver sección 4.3).

Metodología Sistema RCN

Eliminación de píxeles aislados. Después de realizar la detección de bordes y binarización de la imagen, para reducir los errores en la siguiente etapa de reconocimiento de elipses, se eliminan los píxeles que no son, ni serán parte del borde de las elipses (píxeles aislados), utilizando un proceso simple de eliminación. El propósito es desconectar los píxeles de borde de los objetos no circulares de aquellos píxeles de borde que pertenecen a un objeto elíptico. En la figura 3.5, se muestran las posibles posiciones en las cuales se puede encontrar un pixel aislado.

Al analizar los niveles de brillo y contraste en la sección 3.2, observamos que aún cuando se utilizó una técnica de iluminación preestablecida en el sistema de captura, no se pudo obtener la distribución de luz continua para que los niveles de gris en la imagen fuesen homogéneos. Los resultados obtenidos sirvieron para establecer los valores de las constantes y los rangos de la técnica de transformación lineal para realzar el contraste.

			X		
	N		X	N	X
			X		
	X		N		X
	N			X	
X	X	X	X		

Figura 3.6. Eliminación de píxeles aislados

Extracción de regiones. En esta sección se utilizó el primer algoritmo de búsqueda profunda para obtener las regiones de píxeles de borde conectadas entre sí. (Pajares & De la Cruz, 2001), evaluando al extraer cada región si su número de píxeles se encuentra en el rango de valores del parámetro tamaño, con el fin de preseleccionar las regiones que serán evaluadas. Cada una de estas regiones fueron almacenadas en una estructura de datos, donde su índice principal es el nombre de la región y su valor son cuatro coordenadas, que describen los puntos máximos de la posición de la elipse 1) superior, 2) izquierdo, 3) derecho y 4) inferior.

Extracción de parámetros de la elipse. Se implementó el algoritmo aleatorio de la transformada de Hough (ver sección 2.2.6), para obtener los parámetros de la elipse. Este algoritmo fue desarrollado por (McLaughlin, 1998) y ha servido como referencia en la evolución de nuevos algoritmos de reconocimiento como son (Fitzgibbon, 1999), (Zhang, S.C. & Liu, Z.Q, 2004).

Para cada una de las regiones en la imagen, se obtienen los parámetros (p, q, a, b, c) (ver sección 2.2.6) y se verifica que su forma sea elipsoidal, puesto que es posible que en la región exista otra forma. Para realizar dicha verificación, se realizaron los siguientes pasos:

1. Se generaron puntos en el perímetro de la imagen.
2. En cada uno de los puntos generados se evaluó la siguiente condición:

Si el punto generado para la posición (x, y) correspondía con el punto de la imagen en la misma posición (x, y) , se contaba como un punto de la circunferencia de la elipse, en caso contrario no.

3. Se divide el número total de puntos de circunferencia contados entre el valor de la circunferencia de la elipse (ecuación 3.3).

$$c.e = \pi * a * b \quad (3.3)$$

4. Si el valor obtenido en el punto 3 es mayor que 0.87 la elipse existe en la imagen y la validación es positiva.

3.5 Clasificación

De acuerdo a las dimensiones obtenidas se compara el valor del eje o radio menor, en el rango del tamaño de los microsporidios, para las dos metodologías:

N. Ceranae:

$$anchoParticulaProcesada \geq 1.4 \ \&\& \ anchoParticulaProcesada \leq 2$$

N. Apis:

$$anchoParticulaProcesada \geq 2.5 \ \&\& \ anchoParticulaProcesada \leq 4.0$$

4. Discusión de Resultados

En el capítulo anterior se plantearon dos tipos de metodologías que solucionan el problema de reconocimiento y clasificación de los microsporidios de *N. Apis* y *N. Ceranae* mediante técnicas de visión artificial. En este capítulo se entregan los resultados de las comparaciones de las dos metodologías en cada una de sus fases.

4.1 Preprocesamiento de la imagen

Eliminación de ruido

Se aplicó el filtro Gaussiano (ver sección 2.2.1) para ambas metodologías, con el fin de reducir el ruido en cada una de las imágenes. Su resultado no fue el esperado (ver figura 4.1), ya que las imágenes se desenfocaron y empeoró la calidad de cada una de las imágenes respecto a la original. En consecuencia, por la naturaleza de la imagen no tiene ningún sentido considerar un filtro de colores o de eliminación de ruido como el filtro gaussiano, pero se utilizaron otras técnicas para mejorar la calidad de la imagen como se verá a continuación.

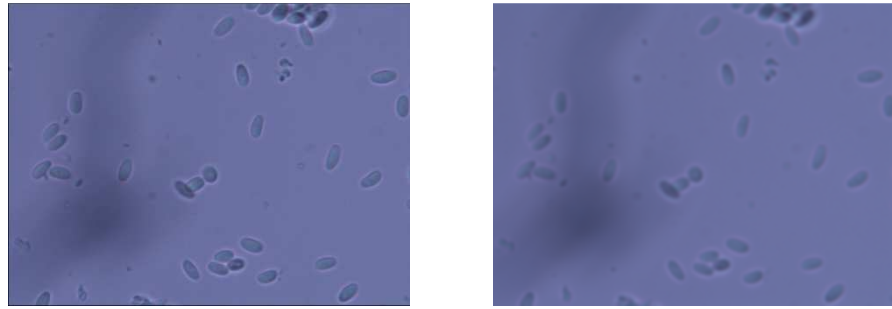


Figura 4.1. Aplicación filtro gaussiano

Transformación Lineal

Se implementó el algoritmo de transformación lineal, planteado en la sección 2.2.3. La constante de parcialidad tomó tres valores (1, 1.5 y 2.0), con el fin de observar la mejora del contraste en la imagen en escala de grises.

Una vez obtenidas las imágenes al utilizar la técnica de transformación lineal (ver figura 4.2), se analizó el histograma de cada una de ellas, con el fin de evaluar si era necesario aplicar un nuevo algoritmo para realzar su contraste.

En los resultados obtenidos aunque la distribución de los niveles de gris presentaron un leve aumento de expansión, no consiguió el necesario para generar una distribución homogénea, haciendo necesario ecualizar el histograma como se describió en la sección 2.2.4.

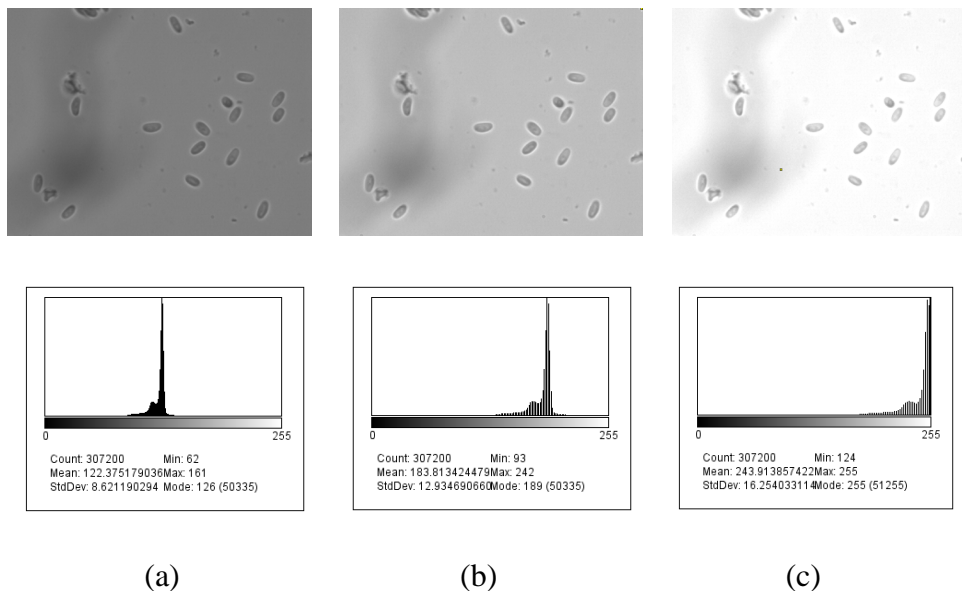


Figura 4.2. Modificación del contraste: (a) imagen original en escala de grises sin modificación del contraste $c=1$; (b) aumento del contraste 50%, $c=1.5$; (c) aumento del contraste en un 100%, $c=2$

Para realzar el contraste de la imagen se seleccionó el algoritmo de ecualización del histograma, el cual fue implementado en dos lenguajes de programación (Java y c#) y se utilizó el método *equalize()* de la plataforma de desarrollo de *ImageJ*, el cual acumula la raíz cuadrada de las entradas del histograma usando un histograma acumulativo (ecuación 3.1), para evitar efectos de contraste extremo.

$$\tilde{H}(i) = \sum_{j=0}^i \sqrt{h(j)} \quad (4.1)$$

En conclusión, después de haber ecualizado las imágenes obtenidas del proceso de transformación lineal, tanto en la implementación del algoritmo clásico de ecualización como en el método *equalize* de la clase *ContrastEnhancer* de la plataforma *ImageJ* y la opción *Histogram Equalization* del software *Image Processing Lab*. Se comprobó que el histograma de intensidades más homogéneo sigue siendo el de la imagen original en escala de grises (ver figura 4.3), por lo tanto el proceso de transformación lineal realizado previamente no genera resultados relevantes en las imágenes al ser ecualizadas para mejorar los niveles de distribución del histograma.

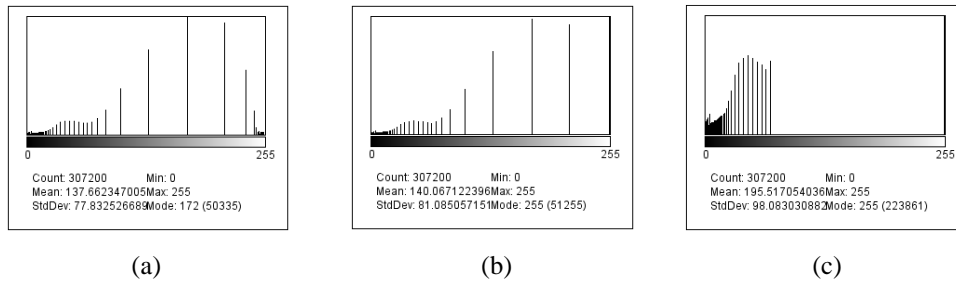


Figura 4.3 Ecualización del Histograma (algoritmo clásico): (a) Imagen original en escala de grises sin modificación del contraste $c=1$ (b) Imagen con contraste del 50%, $c=1.5$ (c) Imagen con aumento del contraste del 100% $c= 2.0$

El algoritmo de ecualización se trato de utilizar en el desarrollo del sistema *RCN* (ver imagen (a) figura 4.4), como se había implementado en el Plugin, pero al realizar la detección de bordes mediante el filtro Sobel, como se puede ver en la imagen (b) de la figura 4.1, se disminuyen las dimensiones de las esporas, ya que toma la membrana que recubre la espora como ruido y lo suaviza, detectando el borde desde la parte interna de la membrana que limita con el centro elipsoidal de la espora.

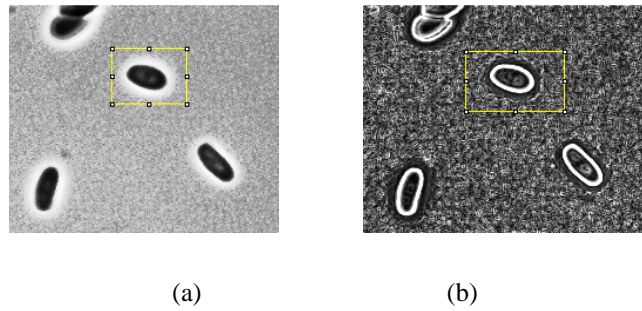


Figura 4.4 Pruebas Sistema RCN. (a) Ecualización de la imagen (b) Detección de bordes algoritmo Sobel

El procedimiento de realce de contraste se aplica solamente al Plugin RCN y es de vital importancia para las fases posteriores de la metodología.

4.2 Segmentación

Plugin RCN. Después de aplicar los algoritmos de umbralización Huang, intermodes, Isodata, Otsu, media (algoritmo implementado), mínimo error, Li y entropía máxima, finalmente se pre-seleccionaron cuatro algoritmos: Otsu, IJ-Isodata, Intermodal y media, por su exactitud al clasificar los esporas del fondo de la imagen y en su proceso de binarización (ver figura 4.5).

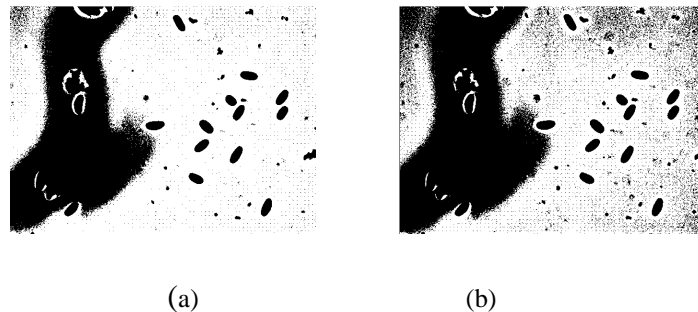


Figura 4.5. Umbralización: (a) Otsu, IJ-Isodata & Intermodal;(b) media

Al aplicar las técnicas de umbralización Otsu, IJ-Isodata e Intermodal, se obtuvieron imágenes similares, casi iguales, a diferencia de los resultados obtenidos con la técnica de umbralización media. A cada una de ellas se le aplicaron los operadores morfológicos de dilatación y erosión (en este mismo orden), obteniendo mejores resultados en las imágenes que se les había aplicado la técnica Otsu y media (imagen (a) y (b), figura 4.6).

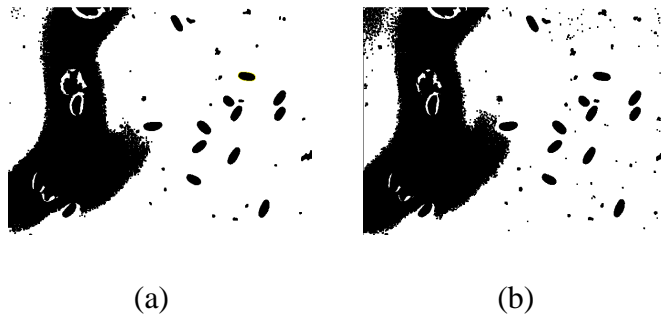


Figura 4.6. Operaciones Morfológicas: (a) con umbralización Otsu y (c) con umbralización filtro medio

Sin embargo, las imágenes que fueron umbralizadas mediante la técnica Otsu pierden píxeles en el perímetro de las esporas, ya que la membrana que la recubre tiene una tonalidad de gris más clara. Debido a esto la técnica fue descartada y se utilizó el filtro medio de umbralización de las librerías *ImageJ*.

Sistema RCN. Después de aplicar los algoritmos de detección de bordes Roberts, Sobel, Prewitt y Canny (ver sección 2.2.4), finalmente se pre-seleccionaron dos algoritmos: Sobel y Canny, por su exactitud en el reconocimiento de bordes de las esporas en el conjunto de imágenes de muestreos (ver figura 4.7).

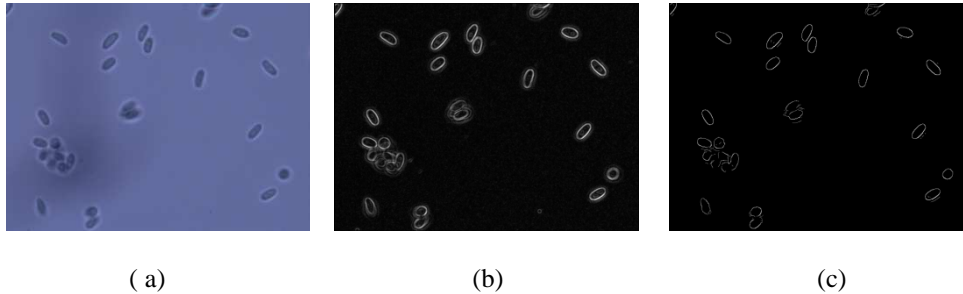


Figura 4.7 Filtros de detección de bordes (a) Imagen original (b) Sobel (c) Canny

Aunque el algoritmo Canny extrajo los bordes y cerró los contornos evitando posibles rupturas de los mismos durante su extracción, el adelgazamiento del ancho de los bordes no nos permitió calcular las dimensiones de las espora correctamente. En consecuencia el operador Canny fue descartado y se utilizó el filtro de detección de bordes Sobel de las librerías de AForge.net.

Para seleccionar la técnica de binarización, se aplicaron al conjunto de muestreo de las imágenes con filtro de detección de bordes Sobel los métodos de umbralización: Isodata, Sys-Threshold y Otsu, de las cuales las técnicas pre-seleccionadas fueron: Otsu e Isodata, por su

precisión en la clasificación de los píxeles de la imagen en dos categorías [0-1], borde o fondo, respectivamente (ver figura 4.8).

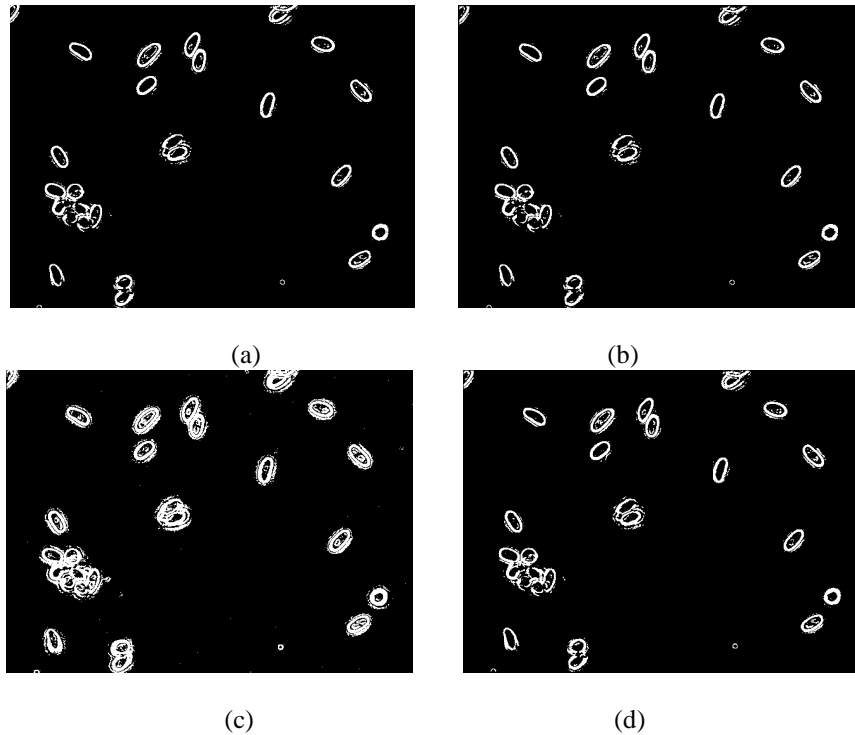


Figura 4.8 Umbralización metodología aplicación (a) Algoritmo Isodata (b) Otsu (c) SysThreshold (d) algoritmo clásico selección automática del umbral

Aunque las técnicas Otsu e Isodata, no presentaron diferencias significativas durante el proceso de binarización, queríamos comprobar que sus resultados fueran mejor que los que obtuvimos implementando el algoritmo iterativo para la selección automática del umbral (Efford, 2000) (ver figura 4.9).

Algoritmo Iterativo para la selección automática del umbral

Computar μ_1 , La media del nivel de gris de los pixeles de las esquinas de la imagen

Computar μ_2 , La media del nivel de gris de los otros pixeles de la imagen.

$T_{anterior} = 0$

$T_{nuevo} = \frac{\mu_1 + \mu_2}{2}$

Mientras $T_{nuevo} \neq T_{anterior}$ entonces haga

μ_1 = La media del nivel de gris de los pixeles donde $f(x, y) < T_{nuevo}$

μ_2 = La media del nivel de gris de los pixeles donde $f(x, y) \geq T_{nuevo}$

$T_{anterior} = T_{nuevo}$

$T_{nuevo} = \frac{\mu_1 + \mu_2}{2}$

Fin Mientras

Figura 4.9. Algoritmo Iterativo para la selección del umbral

Al compara las tres técnicas de binarización obtuvimos que la técnica Otzu presentó mejores resultados. En consecuencia , se escogió esta técnica y se utilizo el filtro Otzu de las librerías de AForge.net.

4.3 Reconocimiento de Formas y dimensiones

Plugin RCN. Es curioso ver que para elaborar un Plugin (ver sección 2.2.8) mediante la plataforma ImageJ, el aspecto más relevante es justamente todo el procesamiento previo a la segmentación de la imagen, ya que una vez la imagen ha sido segmentada con éxito y el rango del tamaño de las partículas ha sido calculado respecto al tamaño de los microsporidios. El modulo analizador de partículas se encarga de extraer las regiones, validar su forma elipsoidal y calcular sus medidas automáticamente.

Sistema RCN. El proceso de extracción de regiones mejoró notablemente al utilizar el rango de valores del parámetro tamaño, excluyendo regiones que no están en el rango de tamaño y que no son objetos de interés (ver figura 4.10).

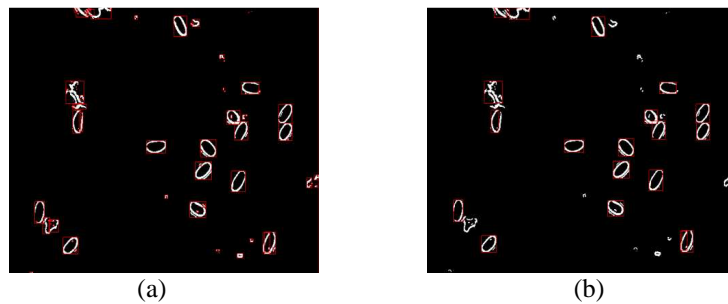


Figura 4.10. Extracción de regiones: (a) sin utilizar parámetro tamaño (b) utilizando parámetro tamaño

La figura 4.11 muestra el resultado de reconocimiento del sistema RCN, de una elipse con orientación de 0° al eje de las x. Naturalmente, las elipses que tienen una orientación de 0° y 90° , se clasifican con éxito si la circunferencia de la elipse se encuentra totalmente cerrada y no tiene pixeles aislados que influyan en el proceso de reconocimiento.

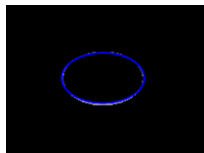


Figura 4.11. Elipse encontrada en el sistema RCN 0°

4.4 Clasificación y Resultados

Para poder medir la efectividad de las metodologías planteadas en el capítulo tres, se utilizaron dos imágenes del conjunto total (diferentes a las imágenes del conjunto de muestra), la primera imagen (I1), se seleccionó del conjunto de imágenes del microsporidio *N. Ceranae* mientras que la segunda imagen (I2) se seleccionó del conjunto de imágenes que tienen los dos microsporidios (*N. Ceranae* y *N. Apis*) (ver figura 4.12).

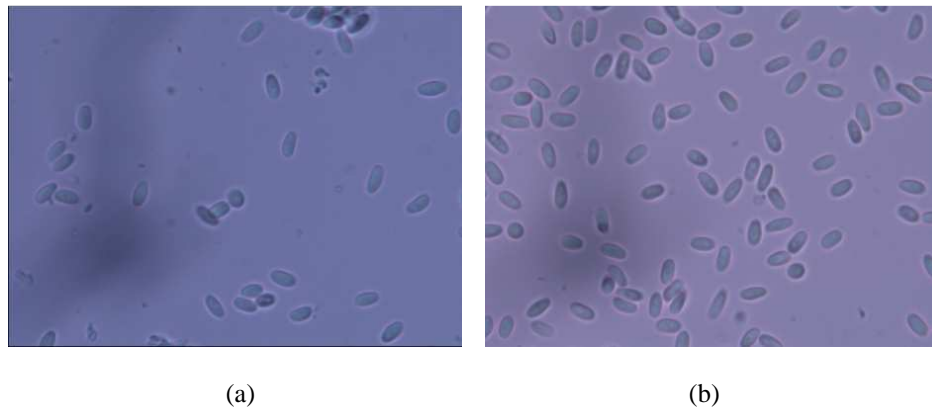


Figura 4.12. Imágenes seleccionadas (a) *N. Ceranae* (b) *N. Ceranae* y *N. Apis*

Procesamiento primera imagen. En la imagen (a) de la figura 4.13, se muestran los resultados obtenidos con el sistema RCN donde los píxeles de borde de los microsporidios 6, 11, 17 y 20, no están completamente conectados entre sí, aun así, los reconoce, mide y clasifica correctamente, por el contrario los microsporidios 1, 2, 4, 7 y 19 fueron clasificados incorrectamente. En el anexo I-tabla 2, se recuperaron los datos del procesamiento de la medición y clasificación del sistema RCN.

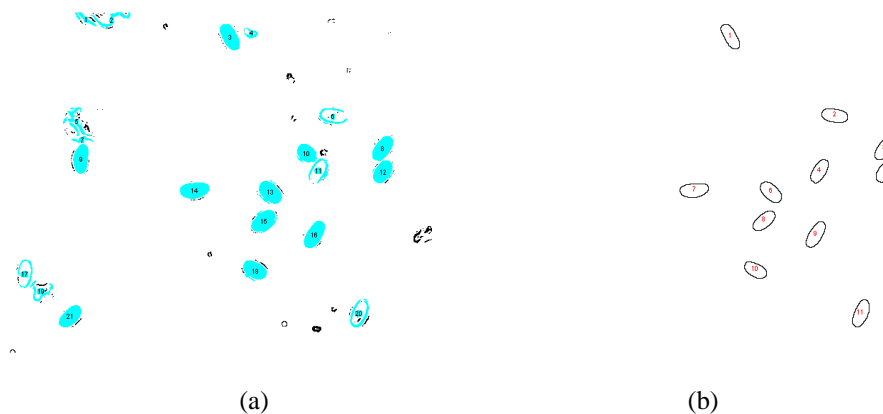


Figura 4.13. Resultados prueba efectividad imagen N1 (a) sistema RCN (b) plugin RCN

Se evaluó la misma imagen para el plugin RCN (ver imagen (b), figura 4.13), consiguiendo resultados de clasificación inferiores a los obtenidos por el sistema (ver tabla 4.1). Sin embargo, aunque no se clasificaron todos los microsporidios tuvo un 52,38% de clasificación exitosa, con 0% de clasificaciones incorrectas. Como se puede ver en las imágenes de la figura 4.12, la parte izquierda inferior de la imagen tiene una coloración más oscura, esto se presento debido a que el filtro del microscopio se encontraba estropeado en esa sección, ocasionando que la metodología de procesamiento de imagen ecualizara esa sección de tal forma que no se pudiese reconocer los microsporidios allí existentes. Esta es una de las causas por la cual la clasificación de las esporas no se realizo. En el anexo I-tabla 1, se recuperaron los datos del procesamiento de la medición y clasificación del plugin RCN.

Metodología	Experto	Clasificación Exitosa	Clasificación Incorrecta	No Clasificadas
Plugin RCN	21(100%)	11 (52,38%)	0(0%)	10 (47,62%)
Sistema RCN	21 (100%)	16 (76,19%)	5 (23,81%)	0(0%)

Tabla 4.1. Resultados de clasificación imagen I1.

Los resultados del cálculo de dimensiones del plugin fueron más acertados que los resultados obtenidos mediante el sistema RCN, esto se debe a que en la etapa de procesamiento y segmentación se busco capturar el centro de la espora sin eliminar o modificar su membrana, de esta forma se obtiene un objeto elipsoidal solido cuya membrana desaparece al fundirse con el fondo, por así decirlo. Mientras que en el sistema se utilizo el detector de bordes Sobel, el cual se encarga de eliminar el ruido suavemente, claro está que en los resultados y pruebas se puede comprobar que en las regiones de interés extraídas, la membrana fue tomada como ruido y se elimino de tal forma que el centro solido de la elipse es más pequeño que el núcleo original.

Metodología	Experto	Clasificación Exitosa	Clasificación Incorrecta	No Clasificada
Plugin RCN				
Total	80(100%)	56(70%)	5(6,25%)	19(23,75%)
N.Apis	21 (13,75%)	10 (47,62%)	1 (4,76%)	10 (47,62%)
N.Ceranae	59 (86,25%)	46 (77,97%)	4 (6,78%)	9 (15, 25%)
Sistema RCN				
Total	80(100%)	48 (60%)	9(11,25%)	23(28,75%)
N.Apis	21 (13,75%)	2 (9,52%)	3 (14,28%)	16(76,19%)
N.Ceranae	59 (86,25%)	46 (77,97%)	6(10,17%)	7(11,86%)

Tabla 4.2. Resultados de clasificación imagen I2.

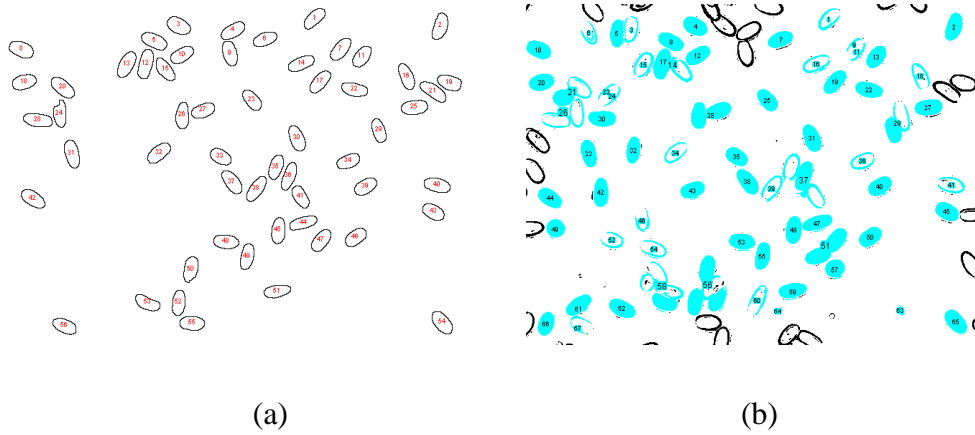


Figura 4.14. Resultados prueba efectividad imagen I2(a) imagen original N-SGE8; (b) plugin RCN; (c) sistema RCN

El plugin obtuvo mejores resultados de medición y clasificación (Exceptuando la parte inferior izquierda de las imágenes), mientras que el sistema obtuvo mejores resultados de reconocimiento y clasificación.

5. Conclusiones y futuros trabajos

Entre las patologías que más afectan a las abejas, la nosemosis ocupa un lugar importante. Sin embargo, en la actualidad, en España son pocos los laboratorios especializados en el estudio de las patologías apícolas, que posean sistemas de ayuda para diagnosticar, este y otros tipos de enfermedades parasitarias.

Es un campo de investigación nuevo y amplio, en el cual se pueden utilizar herramientas de visión artificial, aplicadas a una problemática específica, durante el proceso de investigación, a través del análisis de imágenes microscópicas en el área de las ciencias naturales.

Actualmente existen herramientas robustas para el procesamiento de estas imágenes. Sin embargo, su gran capacidad está restringida por falta de adecuación a la problemática concreta. En esta investigación, se propuso un método simple y directo de reconocimiento, medición y clasificación de microsporidios (*N. Ceranae* y *N. Apis*) usando visión artificial, relacionada a los ámbitos de la Visión por Computador que automatiza el proceso de detección y clasificación de las esporas.

Se consideraron y evaluaron, dos metodologías de tratamiento de imágenes y reconocimiento de patrones de Visión por Computador, con el fin de seleccionar la metodología que presentara los mejores resultados. Para lograr este objetivo se crearon dos nuevas metodologías, cuyo

enfoque era totalmente diferente, desde el procesamiento de las imágenes hasta el sistema de conocimiento y medición. Los resultados obtenidos también variaron notablemente debido a que cada uno de los procesamiento realizados mostraba las fortalezas y debilidades de las técnicas aplicadas y combinadas. A pesar de la continua fluctuación en los resultados obtenidos, la metodología que presento mejores aciertos de medición y clasificación fue el plugin RCN.

Aunque los resultados de los métodos tradicionales utilizados en la detección de *N. Ceranae* basados en análisis microscópicos realizados manualmente, a menudo poseen un alto grado de inexactitud e imprecisión. Siguen presentando mejores resultados, no por mucha diferencia, pero de acuerdo a los sistemas desarrollados en el área, los resultados deben mejorar para ser un sistema fiable.

Se debe evaluar la posibilidad de incorporar otros sistemas expertos, que mejoren el proceso de reconocimiento y clasificación de microsporidios (disminuyendo la variabilidad de la intervención humana).

Aun no es una herramienta diagnóstica, sirve como ayuda en el proceso diagnóstico, y en un futuro será una herramienta que sirva para diagnosticar completamente la enfermedad parasitaria, generando el tratamiento y medicación específica que se debe implantar en la colonia para disminuir la transmisión del microsporidio *N. Ceranae* a la siguiente generación hospedadora.

Utilizando un sistema de gestión de datos, en donde se realizaran los procesos de almacenamiento, procesamiento y validación de la ingente cantidad de datos que aportara el nuevo sistema diagnóstico.

Este nuevo sistema permitirá evaluar la prevalencia de la enfermedad parasitaria, y su evolución.

ANEXO I. TABLAS DE RESULTADOS

Tabla 1. Resultados medición y clasificación plugin RCN imagen N1.

Las seis primeras columnas contienen información de cada una de las regiones de interés, extraídas de la imagen mediante el plugin RCN. Columna 1) Índice 2) Área, 3 y 4)(P-X, P-Y), coordenada de localización (x, y) de la de la región en la imagen, 5y 6) Ancho y alto de la región (rectángulo).

Las columnas 7,8 y 9, contiene información de la elipse de la región. Columna 7) a (radio mayor), 8) b (radio menor) y 9) N (Clasificación: C=Ceranae)

	A	B	C	D	E	F	G	H	I
1	▼ Area ▼	▼ P-X ▼	▼ P-Y ▼	▼ Ancho ▼	▼ Alto ▼	▼ a ▼	▼ b ▼	▼ N ▼	
2	1	535	341	19	26	34	4	1,7	C
3	2	555	482	137	36	19	3,8	1,9	C
4	3	546	556	176	27	32	3,7	1,9	C
5	4	516	466	208	25	33	3,6	1,8	C
6	5	482	558	210	24	30	3,3	1,9	C
7	6	571	396	239	30	29	3,6	2	C
8	7	617	284	242	40	19	4,1	1,9	C
9	8	534	386	280	31	27	3,7	1,9	C
10	9	586	460	295	27	36	4,1	1,8	C
11	10	492	374	351	31	23	3,4	1,8	C
12	11	598	525	404	23	38	4,1	1,9	C

Tabla 2. Resultados medición y clasificación sistema RCN imagen N1.

Las seis primeras columnas contienen información de cada una de las regiones de interés, extraídas de la imagen mediante el sistema RCN. Columna 1) Índice 2) Área, 3 y 4)(P-X, P-Y), coordenada de localización (x, y) de la de la región en la imagen, 5y 6) Ancho y alto de la región (rectángulo).

Las columnas 7,8 y 9, contiene información de la elipse de la región. Columna 7) a (radio mayor), 8) b (radio menor) y 9) N (Clasificación: C=Ceranae o A = Apis).

	A	B	C	D	E	F	G	H	I
1	Area	P-X	P-Y	Ancho	Alto	a	b	N	
2	1	170	136	1	31	20	3,11	0,7	*
3	2	294	160	1	54	23	3,352	1,1	*
4	3	339	340	18	28	37	2,722	1,6	C
5	4	118	374	23	20	15	1,438	1	*
6	5	395	119	134	38	41	3,139	1,6	C
7	6	233	480	136	39	22	2,103	1,4	C
8	7	121	131	175	31	10	2,448	0,6	*
9	8	255	555	175	29	35	2,275	1,4	C
10	9	269	133	186	22	42	2,472	1,4	C
11	10	302	448	186	29	26	2,153	1,8	C
12	11	246	465	206	28	35	1,94	1,6	C
13	12	280	555	209	30	32	2,204	1,6	C
14	13	321	395	236	33	34	2,239	1,8	C
15	14	264	284	240	41	23	2,378	1,4	C
16	15	375	384	278	37	33	2,555	1,9	C
17	16	267	458	294	30	38	2,422	1,4	C
18	17	187	55	348	21	40	2,129	1,1	*
19	18	323	373	349	34	30	2,325	1,8	C
20	19	211	72	377	33	30	1,833	1,5	C
21	20	294	523	403	27	40	2,479	1,5	C
22	21	279	113	412	32	31	2,469	1,4	C

BIBLIOGRAFÍA

Anguita, J. & Pérez. A. *Aplicaciones informáticas para la gestión documental de los sistemas de calidad en el laboratorio clínico*. [http://www.aehh.org/documentos/HAEMAT_PONENCIAS_09_\(01-146\).pdf](http://www.aehh.org/documentos/HAEMAT_PONENCIAS_09_(01-146).pdf), J. haematologica (edición española), Vol. 94 (Extra 1), págs 138-141, 2009.

Burge, W. & Burge, M. J. *Principles of Digital Image Processing, Fundamental Techniques*. (Histograms, págs 52; Point Operations, págs 55- 63), Springer, 2009.

Canny, J. *A computational approach to edge detection*, IEEE Trans. Pattern Analysis and Machine Intelligence, págs 679–714, 1986.

Cerda, M. *Reconocimiento de Bordes en Imágenes, un enfoque aplicado*, Memoria (Antecedentes, págs 6-20), 2007.

Collins, T.J. *ImageJ for microscopy*, J. BioTechniques, vol. 43, No. 1, págs 25-30, 2007.

Coloss, *Nosema disease: lack of knowledge and work standardization*, Proc. Workshop (COST Action FA0803), <http://www.coloss.org/news/nosema-workshop-proceedings-online>, 2009.

Dave, R.N. *Generalized fuzzy c-shells clustering and detection of circular and elliptical boundaries*, Pattern Recognition, Vol. 25, No 7, Págs, 713–721, 1992.

De la Escalera, A. *Visión por computador, fundamentos y métodos*. (Preprocesamiento de imágenes, págs 112-154), Prentice Hall, 2001.

Efford, N. *Digital Image Processing, a practical introduction using Java*. (Segmentation, págs 250-255), Addison-Wesley, 2000.

Fitzgibbon, A.; Pilu, M. & Fisher, R. *Direct Least Square Fitting of Ellipses*, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 21.5, págs 476-480, 1999.

Furundarena, J.R. *Herramientas de ayuda a la validación de resultados en hematimetría*, [http://www.aehh.org/documentos/HAEMAT_PONENCIAS_09_\(01-146\).pdf](http://www.aehh.org/documentos/HAEMAT_PONENCIAS_09_(01-146).pdf), J. haematologica (edición española), Vol. 94 (Extra 1), págs 126-138, 2009.

García, M.P.; Meana, A.; M.; Martín; Sanz, A.; Álvarez, N.; Sanz, A. *El síndrome de despoblamiento de las colmenas en España, consideraciones sobre su origen*. J. Vida apícola, Vol. 133, págs 15-21, 2005.

Gering, E. & Atkinson, C. A Rapid Method for Counting Nucleated Erythrocytes on Stained Blood Smears by Digital Image Analysis, J. Parasitology, No 4, págs 879-881, 2004 .

Girish, V. & Vijayalakshmi, A. Affordable Image Analysis using NIH Image/ImageJ, J. cancer, vol. 41, No 1, pág 47, 2004.

Gonzalez, R. C. & Woods, R. E. *Digital Image Processing*, (Histogram Processing, págs.173-180; Local enhancement, págs. 182-185), Addison-Wesley 1992.

Higes, M.; Martín, R. & Meana, A. *Nosema ceranae in Europe: an emergent type C nosemosis*, Apidologie (in press), 2010.

Igathinathane, C.; Melin, S.; Sokhansanj, S.; Bi, X.; Lim, C.J.; Pordesimo, L.O. and Columbus, E.P. Machine vision based particle size and size distribution determination of airborne dust particles of wood and bark pellets, J. Powder Technology, Vol. 196, No 2, págs 202-212, 2009.

Kirillov, A. AForge.NET Framework, <http://code.google.com/p/aforge/>, 2009.

Masutani, Y.; Schiemann, T. & Höhne, K.-H. Vascular *shape segmentation and structure extraction using a shape based region-growing model*, Proc. MICCAI 98. Lecture Notes in Computer Science, vol. 1496, págs 1242-1249, 1998

McLaughlin R.A. *Randomized hough transform: Improved ellipse detection with comparison*, Pattern Recognition, Vol. 19, págs. 299-305, 1998.

Martín, A.; Meana, A.; García-Palencia, P.; Marín, P.; Botías, C.; Garrido-Bailón, E.; Barrios, L. & Higes, M. *Effect of Temperature on the Biotic Potential of Honeybee Microsporidia*, J. Applied and Environmental Microbiology, Vol. 72, No 8, págs 2554-2557, 2009.

Meana, A.; Martín, R.; Higes, M. *La silenciosa pandemia de las abejas*. M. Investigación y ciencia. Edición Española de Scientific American, págs 8-10, 2009.

Moreno, A.; Marcos, R. *Procesamiento y gestión digital de la información*, Bibl. Univ., Nueva Época, Vol. 9, No 2, págs 144-151, 2006.

Nixon, M. & Aguado, A. *Feature Extraction & Image Processing*. (Filtro gaussiano, pág 98), Elsevier, 2005.

OIE (Office International des Epizooties), *Manual of Standards for Diagnostic Test and Vaccines*, <http://www.oie.int/eng/normes/mmanual/2008/>, 2008.

Otti, O. & Schmid-Hempel, P. *Nosema bombi: A pollinator parasite with detrimental fitness effects*, *J. Invertebr Pathol*, Vol. 96, No. 2, págs 118-124, 2007.

Pajares, G. & De la Cruz, J. *Visión por computador, Imágenes digitales y aplicaciones*. (Suavizado y realzado, págs 96-112; Operaciones Morfológicas, págs 272-278; Extracción de bordes, esquinas y puntos de interés, págs 159 -163; Extracción de regiones, págs. 179-199) , Ra-Ma, 2001.

Rasband, W.S. *ImageJ*, U. S. National Institutes of Health, Bethesda, Maryland, USA, <http://rsb.info.nih.gov/ij/>, 1997-2009.

Sossa, Y.; Osorio, G.; Prieto, F. & Angulo, F. *Conteo y caracterización de regiones completas en imágenes 2D: Aplicación a núcleos celulares*, *Revista Colombiana de Física*, Vol. 34, No 1, 2002.

Stefancik, R. M. & Sonka, M. *Highly automated segmentation of arterial and venous trees from three-dimensional magnetic resonance angiography (MRA)*, *J. Cardiovasc. Imaging.*, vol. 17, págs 37–47, 2001.

Trier and, Ø. D.& Jain, A. K. *Goal-directed evaluation of binarization methods*, *IEEE Trans. Image Proc.*, vol. 17, No. 12, págs. 1191–1201, 1995.

Venkateswarlu, N.B. & Raju, P.S.V.S.K. *Fast isodata clustering algorithms*, Department of Computer Science, Birla Institute of Technology and Science, India, 2003.

Yang, Y. & Yan, H. An adaptive logical method for binarization of degraded document images, *Pattern Recognition*, vol. 33, págs 787–807, 2000.

Yin, P. *A new circle/ellipse detector using genetic algorithms*, *Proceedings of the IPPR on CVGIP Taipei*, págs 362–368, 1998.

Zhang, S.C. & Liu, Z.Q. *A Robust, real-time ellipse detector*. *Pattern Recognition*, Vol. 38, págs 273-287, 2004.

RENUNCIA

La abajo firmante, matriculada en el Máster en Investigación en Informática de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Máster: “CDN: Sistema de Clasificación y Diagnósticos de Nosemosis”, realizado durante el curso académico 2009-2010 bajo la dirección de Matilde Santos Peñas y con la colaboración externa de dirección de Aránzazu Meana de la facultad de Veterinaria, y a la Biblioteca de la UCM a depositar en el archivo institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en internet y garantizar su preservación y acceso a largo plazo